PATENT ABSTRACTS OF JAPAN

(11)Publication number : 11-143902
(43)Date of publication of application : 28.05.1999

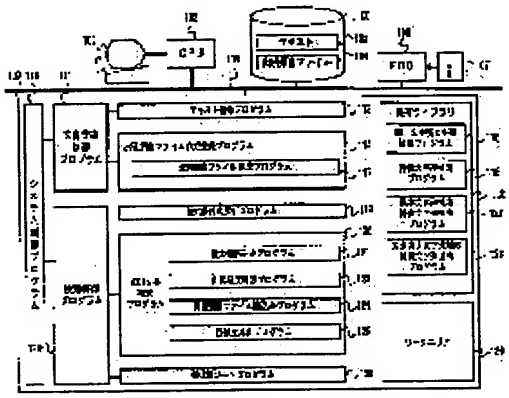
(51)Int.Cl. G06F 17/30

(21)Application number : 09-309078
(22)Date of filing : 11.11.1997

(71)Applicant : HITACHI LTD
(72)Inventor : MATSUBAYASHI TADATAKA
TADA KATSUMI
OKAMOTO TAKUYA
SUGAYA NATSUKO
KAWASHITA YASUSHI

(54) SIMILAR DOCUMENT RETRIEVAL METHOD USING N-GRAM

(57)Abstract:
PROBLEM TO BE SOLVED: To provide a similar document retrieval system which can retrieve even for such languages as the Japanese have many character types at a high speed and with high accuracy.
SOLUTION: This system is provided with a step where the appearance frequency of a feature character string existing in a text 103 contained in a text data base is stored as an appearance frequency file 104, a step where the feature character string is extracted from the text that is designated by a user and a step where the appearance frequency of the feature character string is counted in the text designated by the user. Then, a similarity is calculated to the text designated by the user by using the appearance frequency stored in the file 104 and in the text designated by the user. Thus, the similar documents are retrieved by using the calculated similarity.



LEGAL STATUS
[Date of request for examination] 17.09.2002
[Date of sending the examiner's decision of rejection]
[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]
[Date of final disposal for application]
[Patent number]
[Date of registration]
[Number of appeal against examiner's decision of rejection]
[Date of requesting appeal against examiner's decision of rejection]
[Date of extinction of right]

(19)日本国特許庁（ＪＰ）　(12)公開特許公報（A）　(11)特許出願公開番号

特開平11－143902

(43)公開日　平成11年(1999)５月28日

(51)Int.Cl.[6]	識別記号	ＦＩ	
Ｇ０６Ｆ　17/30		Ｇ０６Ｆ　15/401	３１０Ａ
		15/403	３５０Ｃ

審査請求　未請求　請求項の数10　ＯＬ　（全 27 頁）

(21)出願番号　特願平9－309078

(22)出願日　平成９年(1997)11月11日

(71)出願人　000005108
株式会社日立製作所
東京都千代田区神田駿河台四丁目６番地

(72)発明者　松林　忠孝
神奈川県川崎市幸区鹿島田890番地　株式会社日立製作所情報・通信開発本部内

(72)発明者　多田　勝己
神奈川県川崎市幸区鹿島田890番地　株式会社日立製作所情報・通信開発本部内

(72)発明者　岡本　卓哉
神奈川県川崎市幸区鹿島田890番地　株式会社日立製作所情報・通信開発本部内

(74)代理人　弁理士　小川　勝男

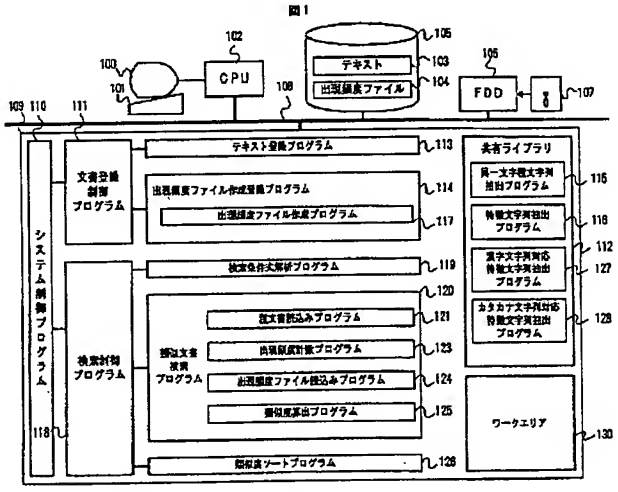
最終頁に続く

(54)【発明の名称】　ｎ－ｇｒａｍを用いた類似文書検索方法

(57)【要約】

【課題】本発明の課題は、日本語のように文字種の多い言語に対しても、高速で高精度な類似文書検索システムを提供することである。

【解決手段】テキストデータベース中のテキスト１０３に存在する特徴文字列のそのテキスト１０３における出現頻度を出現頻度ファイル１０４として格納するステップと、ユーザが指定したテキストから特徴文字列を抽出するステップと、ユーザが指定したテキストにおける特徴文字列の出現頻度を計数するステップとを有し、出現頻度ファイル１０４とユーザが指定したテキストにおける出現頻度を用いてユーザが指定したテキストに対する類似度を算出し、算出された類似度を用いて文書を検索する。



【特許請求の範囲】

【請求項１】文字情報をコードデータとして蓄積したテキストデータベースを対象に、ユーザが指定した文書と類似する文書を検索する類似文書検索方法において、
ユーザが指定した文書のテキスト（指定テキストと呼ぶ）から所定の文字種の変わり目を境界として文字列を抽出する文字列抽出ステップと、
予め定められた一つ以上の文字列の種類に応じて、その中から一つ以上の部分文字列を抽出する検索用部分文字列抽出ステップと、
該指定テキストに対する該テキストデータベース中のテキストの類似度を所定の類似度算出式を用いて算出する類似度算出ステップを有することを特徴とした類似文書検索方法。

【請求項２】請求項１記載の類似文書検索方法における前記文字列抽出ステップで、
該指定テキストから抽出する文字列として、全ての文字種の変わり目を境界として同一文字種からなる文字列を抽出する同一文字種文字列抽出ステップを有することを特徴とした類似文書検索方法。

【請求項３】請求項２記載の類似文書検索方法における前記検索用部分文字列抽出ステップで、
全ての文字種に応じて予め定められた文字列長の部分文字列を検索用部分文字列として抽出する文字種別検索用部分文字列抽出ステップを有することを特徴とした類似文書検索方法。

【請求項４】請求項１、２および３に記載の類似文書検索方法における前記検索用部分文字列抽出ステップで、
予め定められた長さの文字列を検索用部分文字列として抽出するステップ、
前記文字列抽出ステップで抽出された文字列そのものを検索用部分文字列として抽出するステップ、
前記文字列抽出ステップで抽出された文字列とその部分文字列の指定テキストにおける出現頻度比を算出し、所定値を満たす部分文字列を検索用部分文字列として抽出するステップ、
前記文字列抽出ステップで抽出された文字列から、予め作成しておいた、検索用部分文字列として抽出しない文字列を不要語として記載した排除文字列辞書に含まれない文字列を、検索用部分文字列として抽出するステップ、
および、前記文字列抽出ステップで抽出された文字列から検索用部分文字列としては部分文字列を抽出しないステップ、
のいずれか一つ、あるいは、それらを組み合わせることにより検索用部分文字列を抽出する検索用部分文字列抽出ステップを有することを特徴とした類似文書検索方法。

【請求項５】請求項１、２および３に記載の類似文書検索方法における前記検索用部分文字列抽出ステップで、
予め定められた長さの文字列を検索用部分文字列として抽出する所定長文字列抽出ステップ、
前記文字列抽出ステップで抽出された文字列そのものを検索用部分文字列として抽出する最長文字列抽出ステップ、
前記文字列抽出ステップで抽出された文字列とその部分文字列の指定テキストにおける出現頻度比を算出し、所定値を満たす部分文字列を検索用部分文字列として抽出する高出現頻度比文字列抽出ステップ、
上記所定長文字列抽出ステップ、最長文字列抽出ステップおよび高出現頻度比文字列抽出ステップの中の少なくとも一つの抽出ステップで抽出された部分文字列から、予め作成しておいた、検索用部分文字列として抽出しない文字列を不要語として記載した排除文字列辞書に含まれる文字列を削除するステップ、
および、前記文字列抽出ステップで抽出された文字列から検索用部分文字列としては部分文字列を抽出しないステップ、
のいずれか一つ、あるいは、それらを組み合わせることにより検索用部分文字列を抽出する検索用部分文字列抽出ステップを有することを特徴とした類似文書検索方法。

【請求項６】請求項１、２、３および４に記載の類似文書検索方法において、
前記検索用部分文字列抽出ステップで抽出された検索用部分文字列の重要度を、予め定められた算出式を用いて算出し、所定値を満たす検索用部分文字列を抽出する検索用部分文字列選択ステップを有することを特徴とした類似文書検索方法。

【請求項７】請求項５記載の類似文書検索方法における前記検索用部分文字列選択ステップとして、
前記検索用部分文字列抽出ステップにおいて抽出された検索用部分文字列の文字種類、文字列長、テキストデータベース内の出現文書数、指定テキストにおける出現頻度および該テキストにおける出現位置等の情報のいずれか一つ、あるいは、それらを組み合わせて、検索用部分文字列の重要度を算出する重要度算出ステップを有することを特徴とした類似文書検索方法。

【請求項８】請求項６記載の類似文書検索方法において、
登録時に検索用部分文字列のテキストデータベース内の出現文書数を出現文書数ファイルとして保存する出現文書数ファイル作成ステップを有し、
検索時における前記重要度算出ステップにおいて、上記出現文書数ファイルから該検索用部分文字列の出現文書数を読み込む出現文書数ファイル読込みステップを有することを特徴とした類似文書検索方法。

【請求項９】請求項６および７に記載の類似文書検索方法において、
登録時に検索用部分文字列の重要度を予め定められた算

出式を用いて算出し、これを重要度ファイルとして保存する重要度ファイル作成ステップを有し、
検索時における前記重要度算出ステップにおいて、上記重要度ファイルから該検索用部分文字列の重要度を読み込む重要度ファイル読込みステップを有することを特徴とした類似文書検索方法。
【請求項１０】請求項１～８に記載の類似文書検索方法において、
登録時に検索用部分文字列のテキストデータベース内の各テキストにおける出現頻度を出現頻度ファイルとして保存する出現頻度ファイル作成ステップを有し、
検索時における前記類似度算出ステップにおいて、上記出現頻度ファイルから出現頻度情報を読み込む出現頻度ファイル読込みステップを有することを特徴とした類似文書検索方法。
【発明の詳細な説明】
【０００１】
【発明の属する技術分野】本発明は、ユーザが指定した文書と類似する文書を、文書データベースの中から検索する方法に関する。
【０００２】
【従来の技術】近年、パーソナルコンピュータやインターネット等の普及に伴い、電子化文書が爆発的に増加しており、今後も加速度的に増大していくものと予想される。このような状況において、ユーザが所望する情報を含んだ文書を高速かつ効率的に検索したいという要求が高まってきている。
【０００３】このような要求に応える技術として全文検索がある。全文検索では、検索対象文書をテキストとして計算機システムに登録してデータベース化し、この中からユーザが指定した検索文字列（以下、検索タームと呼ぶ）を含む文書を検索する。このように全文検索では、文書中の文字列そのものを対象として検索を行なうため、予めキーワードを付与し、このキーワードを手掛りに検索する従来の検索システムとは異なり、検出漏れが原理的に生じないという特長がある。
【０００４】しかし、ユーザが所望する情報を含んだ文書を的確に検索するためには、ユーザの検索意図を正確に表す複雑な検索条件式を入力する必要がある。
【０００５】この繁雑さを解消するために、ユーザが自分の所望する内容の文書（以下、種文書と呼ぶ）を指定し、その文書と類似する文書を検索する類似文書検索技術が注目されている。
【０００６】類似文書検索の方法としては、例えば、「特開平８－３３５２２２号公報」に、形態素解析等により種文書中に含まれる単語を抽出し、これを用いて類似文書を検索する技術（以下、従来技術１と呼ぶ）が開示されている。
【０００７】また、「特開平６－１１０９４８号公報」には、種文書中から連続するn文字の文字列（以下、n-gramと呼ぶ）を抽出し、これを用いて類似文書を検索する技術（以下、従来技術２と呼ぶ）が開示されている。
【０００８】上記２つの従来技術について、その概要を説明する。
【０００９】従来技術１では、形態素解析により種分書中に含まれる単語を抽出し、この単語を含む文書を類似文書として検索する。例えば、「この装置は地下水脈の観測にも使える。」という文書を種文書とする場合、形態素解析により単語辞書を参照して、「装置」「地下」「水脈」「観測」「使える」という単語が抽出される。この結果、「地下水脈を観測することによる地震の発生を予測する。」という文書を類似文書として検索することができる。　しかし従来技術１では、単語の抽出に単語辞書を用いるため、次のような２つの問題が生じる。
【００１０】まず、単語辞書に含まれていない単語は、種文書から検索用の単語として抽出されないためこの単語を含む文書を検索することができないという問題がある。このため、ユーザが所望する情報が新語で表され、これが単語辞書に含まれていない場合、目的の情報を含む文書を検索することができなくなる。
【００１１】次に、ユーザが所望する情報を表わす言葉が単語辞書に含まれる場合でも、単語の抽出の仕方によっては検索漏れが生じてしまう。例えば、上記の「この装置は地下水脈の観測にも使える。」という種文書からは、「装置」「地下」「水脈」「観測」「使える」という単語が抽出される。しかし、「地下水」という単語が抽出されないため「地下水の大量汲み上げで地盤沈下地域が拡大した。」という文書は、類似文書として検索することができないという問題がある。
【００１２】以上が従来技術１の問題点である。
【００１３】この問題を解決するために、上記従来技術２が提案されている。これは、n-gramを用いた類似文書検索方法である。
【００１４】以下、文書１「新開発の心電計による発作時の心電図」、文書２「新しいソフトウェアの開発作業」、および文書３「ソフト開発を支援するソフトウェア」が登録されているデータベースを対象に、n-gramのnの値を２として、ユーザが文書２を種文書に指定した場合を例に、従来技術２の具体的な処理方法を説明し、その問題点を述べる。
【００１５】まず、データベース中の文書１～文書３から2-gramを抽出する。
【００１６】
【表１】

表１

No.	重複排除2-gram	出現頻度	ウェイト
1-1	新開	1	(1/16)= 0.063
1-2	開発	1	0.063
1-3	発の	1	0.063
1-4	の心	2	(2/16)= 0.125
1-5	心電	2	0.125
1-6	電計	1	0.063
1-7	計に	1	0.063
1-8	によ	1	0.063
1-9	よる	1	0.063
1-10	る発	1	0.063
1-11	発作	1	0.063
1-12	作時	1	0.063
1-13	時の	1	0.063
1-14	電図	1	0.063
	（総出現頻度）	16	

【００１７】表１に、文書１に出現する2-gramの中から重複を排除して抽出した2-gram（以下、重複排除2-gramと呼ぶ）を示す。次に、これらの2-gramの各々に対しウェイトを計算する。このウェイトは各々の2-gramの出現頻度をその文書中に出現する2-gramの重複を含めた総出現頻度で割ることによって求める。ウェイトは各々の2-gramのその文書内における出現割合を表し、この値が大きい2-gramほどその文書に頻繁に出現することを意味する。文書２および文書３に対しても同様の処理を施し、それぞれウェイトを求める。表２および表３にこの処理結果を示す。

【００１８】

【表２】

表２

No.	重複排除2-gram	出現頻度	ウェイト
2-1	新し	1	(1/13)= 0.077
2-2	しい	1	0.077
2-3	いソ	1	0.077
2-4	ソフ	1	0.077
2-5	フト	1	0.077
2-6	トウ	1	0.077
2-7	ウェ	1	0.077
2-8	ェア	1	0.077
2-9	アの	1	0.077
2-10	の開	1	0.077
2-11	開発	1	0.077
2-12	発作	1	0.077
2-13	作業	1	0.077
	（総出現頻度）	13	

【００１９】

【表３】

表３

No.	重複排除2-gram	出現頻度	ウェイト
3-1	ソフ	2	(1/15)= 0.133
3-2	フト	2	0.133
3-3	ト開	1	(1/13)= 0.067
3-4	開発	1	0.067
3-5	発を	1	0.067
3-6	を支	1	0.067
3-7	支援	1	0.067
3-8	援す	1	0.067
3-9	する	1	0.067
3-10	るソ	1	0.067
3-11	トウ	1	0.067
3-12	ウエ	1	0.067
3-13	ェア	1	0.067
	（総出現頻度）	15	

【００２０】その後、データベース中の文書間の共通性を除去する。ここでは、まず、データベース中に存在する2-gramの中で重複を排除した2-gramについて、その共通性ウェイトを算出する。この共通性ウェイトは、データベース中の全文書に関する各2-gramのウェイトの総和を、全文書数で割ることによって算出する。共通性ウェイトは、各2-gramのデータベース全体における出現割合を表し、この値が大きい2-gramほどデータベース中のどの文書にも共通して出現することを意味する。

【００２１】

【表４】

表４

No.	重複排除2-gram	ウェイト			共通性ウェイト	
		文　1	文　2	文書３		
4-1	新開	0.063			(0.063+0.0+0.0)/3=	0.021
4-2	開発	0.063	0.077	0.067		0.069
4-3	発の	0.063				0.021
4-4	の心	0.125				0.042
4-5	心電	0.125				0.042
4-6	電計	0.063				0.021
4-7	計に	0.063				0.021
4-8	によ	0.063				0.021
4-9	よる	0.063				0.021
4-10	る発	0.063				0.021
4-11	発作	0.063	0.077			0.046
4-12	作時	0.063				0.021
4-13	時の	0.063				0.021
4-14	電図	0.063				0.021
4-15	新し		0.077			0.026
4-16	しい		0.077			0.026
4-17	いソ		0.077			0.026
4-18	ソフ		0.077	0.133		0.070
4-19	フト		0.077	0.133		0.070
4-20	トウ		0.077	0.067		0.048
4-21	ウェ		0.077	0.067		0.048
4-22	エア		0.077	0.067		0.048
4-23	アの		0.077			0.026
4-24	の開		0.077			0.026
4-25	作業		0.077			0.026
4-26	ト開			0.067		0.022
4-27	発を			0.067		0.022
4-28	を支			0.067		0.022
4-29	支援			0.067		0.022
4-30	援す			0.067		0.022
4-31	する			0.067		0.022
4-32	るソ			0.067		0.022

【００２２】表４に、文書１、文書２および文書３の間の共通性ウェイトを示す。

【００２３】例えば、2-gram「新開」の共通性ウェイトは、(0.063+0.0+0.0)/3=0.021である。ここで2-gram「新開」は文書２および文書３に出現していないのでウェイトはそれぞれ「0.0」となっている。2-gram「開発」の共通性ウェイトは、(0.063+0.077+0.067)/3=0.069である。

【００２４】上述したように、共通性ウェイトは各n-gramのウェイトの平均値である。

【００２５】この共通性ウェイトを各n-gramのウェイトから差し引くことにより、データベース中の文書間の共通性を除去する（この値を従来技術２では、正規化ウェイトと呼んでいる）。正規化ウェイトは、データベースにおける各n-gramの出現偏りを表し、この値が大きいn-gramほどある特定の文書に偏って出現することを意味する。

【００２６】もし、あるn-gramが全ての文書に同じ割合で出現していれば、ウェイトと共通性ウェイトは同じ値となるため、正規化ウェイトは「０」となる。つまり、どの文書においても同じような割合で出現するn-gramに関しては、ウェイトが限りなく「０」に近づくことになる。

【００２７】表５、表６および表７に、文書１、文書２および文書３の正規化ウェイトを示す。

【００２８】

【表５】

表５

No.	重複排除2-gram	ウェイト	共通性ウェイト	正規化ウェイト	
5-1	新開	0.063	0.021	(0.063-0.021)=	0.042
5-2	開発	0.063	0.069		-0.006
5-3	発の	0.063	0.042		0.021
5-4	の心	0.125	0.042		0.083
5-5	心電	0.125	0.021		0.104
5-6	電計	0.063	0.021		0.042
5-7	計に	0.063	0.021		0.042
5-8	によ	0.063	0.021		0.042
5-9	よる	0.063	0.021		0.042
5-10	る発	0.063	0.021		0.042
5-11	発作	0.063	0.046		0.016
5-12	作時	0.063	0.021		0.042
5-13	時の	0.063	0.021		0.042
5-14	電図	0.063	0.021		0.042

【００２９】

【表６】

表６

No.	重複排除 2-gram	ウェイト	共通性 ウェイト	正規化ウェイト	
6-1	新し	0.077	0.026	(0.063-0.026)=	0.051
6-2	しい	0.077	0.026		0.051
6-3	いソ	0.077	0.026		0.051
6-4	ソフ	0.077	0.070		0.007
6-5	フト	0.077	0.070		0.007
6-6	トウ	0.077	0.048		0.029
6-7	ウェ	0.077	0.048		0.029
6-8	ェア	0.077	0.048		0.029
6-9	アの	0.077	0.026		0.051
6-10	の開	0.077	0.026		0.051
6-11	開発	0.077	0.069		0.008
6-12	発作	0.077	0.046		0.030
6-13	作業	0.077	0.026		0.051

【００３０】

【表７】

表７

No.	重複排除 2-gram	ウェイト	共通性 ウェイト	正規化ウェイト	
7-1	ソフ	0.133	0.070	(0.133-0.070)=	0.063
7-2	フト	0.133	0.070		0.063
7-3	ト開	0.067	0.022		0.044
7-4	開発	0.067	0.069		-0.002
7-5	発を	0.067	0.022		0.044
7-6	を支	0.067	0.022		0.044
7-7	支援	0.067	0.022		0.044
7-8	援す	0.067	0.022		0.044
7-9	する	0.067	0.022		0.044
7-10	るソ	0.067	0.022		0.044
7-11	トウ	0.067	0.048		0.019
7-12	ウェ	0.067	0.048		0.019
7-13	ェア	0.067	0.048		0.019

【００３１】以上のようにして得られた正規化ウェイトを用いて、ユーザが種文書として指定した文書とデータベース中の全文書との類似性を求め、これを類似度として表わす。文書番号をiとすると、文書iの類似度S(i)は、以下に示す式（１）によって求められる。

【００３２】

【数１】

数式１

$$S(i)=\frac{\sum_{j=1}^{n}(U(j)\times R(j))}{\sqrt{\sum_{j=1}^{n}U(j)^2\times\sum_{i=1}^{n}R(j)^2}} \quad \cdots \quad 1$$

【００３３】ここで、U(j)は種文書中のj番目のn-gramの正規化ウェイトを示し、R(j)はデータベース中文書のj番目のn-gramの正規化ウェイトを示す。また、nはデータベース中の全文書数を表わす。この式を用いてデータベース中の全ての文書の類似度を算出すると以下のようになる。

【００３４】

S(1) = 0.018

S(2) = 1.0

S(3) = 0.119

最後に、得られた類似度の降順に文書を出力する。この例では、文書２、文書３、文書１の順で出力されることになる。

【００３５】以上が、従来技術２の具体的な処理内容である。このように従来技術２では、単語辞書に基づく形態素解析を用いることなく種文書に類似する文書を検索することができるため、従来技術１における２つの問題点を解決することができる。

【００３６】しかし、この従来技術２には次のような２つの問題点がある。

【００３７】まず、第一の問題点は、種文書から抽出されるn-gram数が膨大になるため、検索に長大な時間を要してしまうという問題である。例えば、1,000文字からなる種文書から全ての2-gramを抽出した場合、999個の2-gramが抽出されることになる。そのため、抽出した全ての2-gramを類似検索に用いる従来技術２の方法では、1個の2-gramの検索が0.1秒で済んだとしても、999個の2-gramでは99.9秒、すなわち約1分40秒も検索時間が掛かってしまうことになる。

【００３８】また、第二の問題点は、全てのn-gramを用いて類似文書を検索するため、検索結果にノイズが含まれるという問題である。

【００３９】以下、この問題点を、文書１～文書３が登録されている前記データベースに、文書４「新しいソフトクリーム券の配布作業」を追加した場合を例に、具体的に説明する。

【００４０】本例では、文書２が、種文書としてユーザ

に指定されたものとする。

【００４１】まず、文書４から2-gramを抽出し、ウェイトを求めた結果を表８に示す。

【００４２】

【表８】

表8

No.	重複排除2-gram	出現頻度	ウェイト
8-1	新し	1	(1/15)= 0.067
8-2	しい	1	0.067
8-3	いソ	1	0.067
8-4	ソフ	1	0.067
8-5	フト	1	0.067
8-6	トク	1	0.067
8-7	クリ	1	0.067
8-8	リー	1	0.067
8-9	ーム	1	0.067
8-10	ム券	1	0.067
8-11	券の	1	0.067
8-12	の配	1	0.067
8-13	配付	1	0.067
8-14	付作	1	0.067
8-15	作業	1	0.067
	（総出現頻度）	15	

【００４３】この文書４のウェイトと表１～表３に示した文書１～文書３のウェイトを用いて、共通性ウェイトを算出する。

【００４４】

【表９】

表9

No.	重複排除2-gram	ウェイト				共通性ウェイト
		文書1	文書2	文書3	文書4	
9-1	新開	0.063				0.016
9-2	開発	0.063	0.077	0.067		0.052
9-3	発の	0.063				0.016
9-4	の心	0.125				0.031
9-5	心電	0.125				0.031
9-6	電計	0.063				0.016
9-7	計に	0.063				0.016
9-8	によ	0.063				0.016
9-9	よる	0.063				0.016
9-10	る発	0.063				0.016
9-11	発作	0.063	0.077			0.035
9-12	作時	0.063				0.016
9-13	時の	0.063				0.016
9-14	電図	0.063				0.016
9-15	新し		0.077		0.067	0.036
9-16	しい		0.077		0.067	0.036
9-17	いソ		0.077		0.067	0.036
9-18	ソフ		0.077	0.133	0.067	0.069
9-19	フト		0.077	0.133	0.067	0.069
9-20	トウ		0.077	0.067		0.036
9-21	ウェ		0.077	0.067		0.036
9-22	ェア		0.077	0.067		0.036
9-23	アの		0.077			0.019
9-24	の開		0.077			0.019
9-25	作業		0.077		0.067	0.036
9-26	ト開			0.067		0.017
9-27	発を			0.067		0.017
9-28	を支			0.067		0.017
9-29	支援			0.067		0.017
9-30	援す			0.067		0.017
9-31	する			0.067		0.017
9-32	るソ			0.067		0.017
9-33	トク				0.067	0.017
9-34	クリ				0.067	0.017
9-35	リー				0.067	0.017
9-36	ーム				0.067	0.017
9-37	ム券				0.067	0.017
9-38	券の				0.067	0.017
9-39	の配				0.067	0.017
9-40	配布				0.067	0.017
9-41	布作				0.067	0.017

【００４５】表９に、文書１～文書４の間の共通性ウェイトを示す。例えば、2-gram「開発」の共通性ウェイトは、(0.063+0.077+0.067+0.000)/4=0.052となる。次に、この共通性ウェイトを各文書の重複排除2-gramのウェイトから差し引くことにより、データベース中の文書間の共通性を除去した正規化ウェイトを求める。

【００４６】　【表１０】

表１０

No.	重複排除 2-gram	ウェイト	共通性 ウェイト	正規化ウェイト	
10-1	新開	0.063	0.016	(0.063-0.016)=	0.047
10-2	開発	0.063	0.068		-0.006
10-3	発の	0.063	0.016		0.047
10-4	の心	0.125	0.031		0.094
10-5	心電	0.125	0.031		0.094
10-6	電計	0.063	0.016		0.047
10-7	計に	0.063	0.016		0.047
10-8	によ	0.063	0.016		0.047
10-9	よる	0.063	0.016		0.047
10-10	る発	0.063	0.016		0.047
10-11	発作	0.063	0.035		0.028
10-12	作時	0.063	0.016		0.047
10-13	時の	0.063	0.016		0.047
10-14	電図	0.063	0.016		0.047

【００４７】　【表１１】

表１１

No.	重複排除 2-gram	ウェイト	共通性 ウェイト	正規化ウェイト	
11-1	新し	0.077	0.036	(0.077-0.036)=	0.041
11-2	しい	0.077	0.036		0.041
11-3	いソ	0.077	0.036		0.041
11-4	ソフ	0.077	0.069		0.008
11-5	フト	0.077	0.069		0.008
11-6	トウ	0.077	0.036		0.041
11-7	ウェ	0.077	0.036		0.041
11-8	エア	0.077	0.036		0.041
11-9	アの	0.077	0.019		0.058
11-10	の開	0.077	0.036		0.041
11-11	開発	0.077	0.068		0.009
11-12	発作	0.077	0.035		0.042
11-13	作業	0.077	0.036		0.041

【００４８】　【表１２】

表１２

No.	重複排除 2-gram	ウェイト	共通性 ウェイト	正規化ウェイト	
12-1	ソフ	0.133	0.069	(0.133-0.069)=	0.064
12-2	フト	0.133	0.069		0.064
12-3	ト開	0.067	0.017		0.050
12-4	開発	0.067	0.068		-0.002
12-5	発を	0.067	0.017		0.050
12-6	を支	0.067	0.017		0.050
12-7	支援	0.067	0.017		0.050
12-8	援す	0.067	0.017		0.050
12-9	する	0.067	0.017		0.050
12-10	るソ	0.067	0.017		0.050
12-11	トウ	0.067	0.036		0.031
12-12	ウェ	0.067	0.036		0.031
12-13	エア	0.067	0.036		0.031

【００４９】　【表１３】

表１３

No.	重複排除2-gram	ウェイト	共通性ウェイト	正規化ウェイト
13-1	新し	0.067	0.036	(0.067-0.036)= 0.031
13-2	しい	0.067	0.036	0.031
13-3	いソ	0.067	0.036	0.031
13-4	ソフ	0.067	0.069	-0.003
13-5	フト	0.067	0.069	-0.003
13-6	トク	0.067	0.017	0.050
13-7	クリ	0.067	0.017	0.050
13-8	リー	0.067	0.017	0.050
13-9	ーム	0.067	0.017	0.050
13-10	ム券	0.067	0.017	0.050
13-11	券の	0.067	0.017	0.050
13-12	の配	0.067	0.017	0.050
13-13	配布	0.067	0.017	0.050
13-14	布作	0.067	0.017	0.050
13-15	作業	0.067	0.036	0.031

【００５０】表１０～表１３に文書１～文書４における2-gramの正規化ウェイトを示す。これらを用いて、種文書である文書２に対する各文書の類似度を式（１）を用いて算出すると、

S(1) = 0.036

S(2) = 1.0

S(3) = 0.179

S(4) = 0.190

となる。

【００５１】ここで、文書３は文書２と同様にソフトウェアに関する文書であるにも関わらず、関係のない文書４の方が文書２に類似していると判断されてしまっている。これは、文書２の「ソフトウェア」から抽出される「ソフ」「フト」が、文書４の「ソフトクリーム」からも抽出されることによる。n-gramは単語のように意味的にまとまった単位の文字列ではないため、同じn-gramであっても文書内で同じ意味を表現しているとは限らない。そのため、この例のように全く関係のない文書が高い類似度を持つ文書として探し出されてしまうという問題がある。

【００５２】

【発明が解決しようとする課題】こうした従来技術の問題に対し、本発明では以下の課題を解決することを目的とする。

【００５３】（１）検索精度の高い類似文書検索方法を提供する。

【００５４】（２）日本語のように文字種の多い言語に対しても、高速に類似文書検索が行える方法を提供する。

【００５５】

【課題を解決するための手段】上記課題を解決するために、本発明による文書検索方法では、以下に示すステップで種文書と類似する文書を検索する。

【００５６】すなわち、本発明による文書検索方法では、文書の登録処理として、（ステップ１）登録対象文書を読み込む文書読込みステップ、（ステップ２）上記文書読込みステップで読み込んだ登録対象文書の文字列を、漢字やカタカナ等の文字種境界で分割し、同一文字種で構成される文字列（以下、同一文字種文字列と呼ぶ）として抽出する同一文字種文字列抽出ステップ、（ステップ３）上記同一文字種文字列抽出ステップで抽出した同一文字種文字列に対して、その文字種を判定し、漢字ならば予め定められた長さの文字列を自立語の可能性があるもの（以下、特徴文字列と呼ぶ）として、そこから抽出し、カタカナや英字ならば同一文字種文字列そのものを特徴文字列として抽出し、それ以外の文字種ならば特徴文字列としては抽出を行わない登録用特徴文字列抽出ステップ、（ステップ４）上記登録用特徴文字列抽出ステップで抽出した特徴文字列に関して、登録対象文書内における出現頻度を計数する出現頻度計数ステップ、（ステップ５）上記出現頻度計数ステップで計数した出現頻度を該当する出現頻度ファイルに格納する出現頻度ファイル作成登録ステップ、を有し、種文書に類似する文書の検索処理として、（ステップ６）種文書を読み込む種文書読込みステップ、（ステップ７）上記種文書読込みステップにおいて読み込んだ種文書の文字列を文字種境界で分割し、同一文字種文字列として抽出する同一文字種文字列抽出ステップ、（ステップ８）上記同一文字種文字列抽出ステップで抽出した同一文字種文字列に対して、その文字種を判定し、漢字ならば予め定められた長さの文字列を特徴文字列としてそこから抽出し、カタカナや英字ならば同一文字種文字列そのものを特徴文字列として抽出し、それ以外の文字種ならば特徴文字列としては抽出を行わない検索用特徴文字列抽出ステップ、（ステップ９）上記検索用特徴文字列抽出ステップで抽出した特徴文字列に関して、種文書内の出現頻度を計数する出現頻度計数ステップ、（ステップ１０）上記出現頻度計数ステップで抽出した全ての特徴文字列に対して、前記出現頻度ファイルを読み込み、データベース内の各文書における出現頻度を取得する出現頻度取得ステップ、（ステップ１１）上記出現頻度取得ステップで抽出した特徴文字列に関し、上記出現頻度計数ステップで計数した種文書内の出現頻度と、上記出現頻度取得ステップで取得したデータベース内の各文書にお

ける出現頻度を用いて、予め定められた算出式に基づいて種文書とデータベース内の各文書との類似度を算出する類似度算出ステップ、（ステップ１２）上記類似度算出ステップで算出した類似度の降順に、文書の一覧を表示する検索結果表示ステップを有する。

【００５７】上記文書検索方法を用いた本発明の原理を、以下に説明する。

【００５８】文書を登録する際には、（ステップ１）～（ステップ５）を実行する。まず、（ステップ１）で登録対象となる文書を読み込む。次に、（ステップ２）において、（ステップ１）で読み込んだ登録対象文書中の文字列を、漢字やカタカナ等の文字種境界で分割し、同一文字種からなる文字列を抽出する。例えば、前記の文書４「新しいソフトクリーム券の配布作業」という文書からは、「新」「しい」「ソフトクリーム」「券」「の」「配布作業」という６個の同一文字種文字列が抽出される。

【００５９】次に、（ステップ３）において、（ステップ２）で抽出した同一文字種文字列について、その文字種を判定し、漢字ならば予め定められた長さの文字列を特徴文字列としてそこから抽出し、カタカナや英字ならば同一文字種文字列そのものを特徴文字列として抽出し、それ以外の文字種ならば特徴文字列としては抽出を行わない。例えば、予め漢字文字列から2-gramを抽出するものと定められている場合には、上記（ステップ２）における同一文字種文字列からは、「ソフトクリーム」「配布」「布作」「作業」が特徴文字列として抽出される。

【００６０】次に、（ステップ４）において、（ステップ３）で抽出した特徴文字列の登録対象文書内における出現頻度を計数する。例えば、上記の文書４「新しいソフトクリーム券の配布作業」という文書では、特徴文字列「ソフトクリーム」が１回出現し、「作業」は１回出現するという情報が得られる。

【００６１】次に、（ステップ５）において、先に（ステップ４）で計数した特徴文字列の出現頻度を該当する出現頻度ファイルに格納する。図２に出現頻度ファイルの例を示す。本図に示した出現頻度ファイルは、表１、表２、表３および表８に示した文書１～文書４を登録した場合の例である。

【００６２】検索時には、（ステップ６）～（ステップ１２）からなる類似文書検索ステップを実行する。

【００６３】まず、（ステップ６）において、種文書として文書２を読み込む。

【００６４】次に、（ステップ７）において、（ステップ６）で読み込んだ種文書（文書２）の文字列を文字種境界で分割し、同一文字種文字列を抽出する。

【００６５】次に、（ステップ８）において、上記（ステップ７）で抽出した同一文字種文字列から、登録時の（ステップ３）と同様の方法で特徴文字列を抽出する。図３に文書２が種文書として指定された場合の（ステップ８）の特徴文字列抽出処理の概要を示す。本図では、同一文字種文字列が漢字の場合には、2-gramを抽出するものとしている。文書２から全ての2-gramを抽出した場合には、１３種類の2-gramが抽出されていたのに対し、本方法では、「ソフトウェア」「開発」「発作」「作業」の４種類の特徴文字列に削減することができている。このように、全てのn-gramを抽出する前述した従来技術２に比べ、本発明では抽出する特徴文字列の種類を大幅に削除できることになる。

【００６６】次に、（ステップ９）において、（ステップ８）で抽出した特徴文字列の種文書内における出現頻度を計数する。そして、（ステップ１０）において、（ステップ８）で抽出した特徴文字列に関して、前述した出現頻度ファイルを参照し、データベース内の各文書における出現頻度を得る。そして、（ステップ１１）において、（ステップ８）で抽出した特徴文字列に対して、（ステップ９）と（ステップ１０）で計数した種文書内における出現頻度と、データベース内の各文書における出現頻度を基に、類似度を算出する。類似度の算出式には、従来技術２で示した式（１）を用いてもよい。式（１）を用いて、文書２が種文書として指定された場合の類似度を算出すると、次のようになる。

【００６７】

$S(1)=0.077$

$S(2)=1.0$

$S(3)=0.263$

$S(4)=0.148$

この結果、（ステップ１２）で、文書を類似度の降順に表示すると、文書２、文書３、文書４および文書１の順に表示される。この類似度算出結果（$S(1)=0.077$、$S(2)=1.0$、$S(3)=0.263$、$S(4)=0.148$）は、従来技術２による類似度算出結果（$S(1)=0.036$、$S(2)=1.0$、$S(3)=0.179$、$S(4)=0.190$）とは異なり、文書２に類似した順に、類似度が正しく算出されることになる。

【００６８】以上のように、本発明の類似文書検索方法によれば、分かち書きのない日本語のような文書に対して、類似文書検索を行なっても、従来技術１のような単語辞書を用いることなく種文書から文字列を機械的に抽出するため、従来技術２のようにどんな単語についても漏れのない検索を行なうことが可能となる。また、従来技術２のように文書中から単純にn-gramを抽出するのではなく、文字種に応じて特徴文字列を抽出することにより、意味のまとまった文字列を用いて検索を行なうことができるため、高精度な類似文書検索を実現することができるようになる。さらに、全n-gramを抽出する従来技術２に比べ、抽出する文字列の種類が大幅に削減されるため、高速に類似文書を検索することができるようになる。

【００６９】

【発明の実施の形態】以下、本発明の第一の実施例について図１を用いて説明する。

【００７０】本発明を適用した類似文書検索システムの第一例は、ディスプレイ１００、キーボード１０１、中央演算処理装置（ＣＰＵ）１０２、磁気ディスク装置１０５、フロッピディスクドライブ（ＦＤＤ）１０６、主メモリ１０９およびこれらを結ぶバス１０８から構成される。

【００７１】磁気ディスク装置１０５は二次記憶装置の一つであり、テキスト１０３、出現頻度ファイル１０４が格納される。ＦＤＤ１０６を介してフロッピディスク１０７に格納されている情報が、主メモリ１０９あるいは磁気ディスク装置１０５へ読み込まれる。

【００７２】主メモリ１０９には、システム制御プログラム１１０、文書登録制御プログラム１１１、共有ライブラリ１１２、テキスト登録プログラム１１３、出現頻度ファイル作成登録プログラム１１４、検索制御プログラム１１８、検索条件式解析プログラム１１９、類似文書検索プログラム１２０および類似度ソートプログラム１２６が格納されるとともにワークエリア１３０が確保される。

【００７３】共有ライブラリ１１２は、同一文字種文字列抽出プログラム１１５、特徴文字列抽出プログラム１１６、漢字文字列対応特徴文字列抽出プログラム１２７およびカタカナ文字列対応特徴文字列抽出プログラム１２８で構成される。

【００７４】出現頻度ファイル作成登録プログラム１１４は、出現頻度ファイル作成プログラム１１７で構成されると共に、後述するように同一文字種文字列抽出プログラム１１５と特徴文字列抽出プログラム１１６を呼び出す構成をとる。

【００７５】類似文書検索プログラム１２０は、種文書読込みプログラム１２１、同一文字種文字列抽出プログラム１１５、出現頻度計数プログラム１２３、出現頻度ファイル読込みプログラム１２４および類似度算出プログラム１２５で構成されると共に、後述するように特徴文字列抽出プログラム１１６を呼び出す構成をとる。

【００７６】文書登録制御プログラム１１１および検索制御プログラム１１８は、ユーザによるキーボード１０１からの指示に応じてシステム制御プログラム１１０によって起動され、それぞれテキスト登録プログラム１１３および出現頻度ファイル作成登録プログラム１１４の制御と、検索条件式解析プログラム１１９、類似文書検索プログラム１２０および類似度ソートプログラム１２６の制御を行なう。

【００７７】以下、本実施例における類似文書検索システムの処理手順について説明する。

【００７８】まず、システム制御プログラム１１０の処理手順について図４のＰＡＤ（Ｐｒｏｂｌｅｍ　Ａｎａｌｙｓｉｓ　Ｄｉａｇｒａｍ）図を用いて説明する。

【００７９】システム制御プログラム１１０は、まずステップ４００で、キーボード１０１から入力されたコマンドを解析する。

【００８０】そしてステップ４０１で、この結果が登録実行のコマンドであると解析された場合には、ステップ４０２で文書登録制御プログラム１１１を起動して、文書の登録を行なう。

【００８１】またステップ４０３で、検索実行のコマンドであると解析された場合には、ステップ４０４で検索制御プログラム１１８を起動して、類似文書の検索を行なう。

【００８２】以上が、システム制御プログラム１１０の処理手順である。

【００８３】次に、図４に示したステップ４０２でシステム制御プログラム１１０により起動される文書登録制御プログラム１１１の処理手順について、図５のＰＡＤ図を用いて説明する。

【００８４】文書登録制御プログラム１１１は、まずステップ５００でテキスト登録プログラム１１３を起動し、ＦＤＤ１０６に挿入されたフロッピディスク１０７から登録すべき文書のテキストデータをワークエリア１３０に読み込み、これをテキスト１０３として磁気ディスク装置１０５に格納する。テキストデータは、フロッピディスク１０７を用いて入力するだけに限らず、通信回線やＣＤ－ＲＯＭ装置（図１には示していない）等を用いて他の装置から入力するような構成を取ることも可能である。

【００８５】次に、ステップ５０１で出現頻度ファイル作成登録プログラム１１４を起動し、磁気ディスク装置１０５に格納されているテキスト１０３を読み出し、その中の各文書における出現頻度ファイル１０４を作成し、磁気ディスク装置１０５に格納する。

【００８６】以上が、文書登録制御プログラム１１１の処理手順である。

【００８７】次に、図５に示したステップ５０１で文書登録制御プログラム１１１により起動される出現頻度ファイル作成登録プログラム１１４の処理手順について、図６のＰＡＤ図を用いて説明する。

【００８８】出現頻度ファイル作成登録プログラム１１４は、まずステップ６００で同一文字種文字列抽出プログラム１１５を起動し、テキスト１０３をワークエリア１３０に読み込み、文字種境界でその文字列を分割することにより同一文字種文字列を抽出し、ワークエリア１３０に格納する。

【００８９】次に、ステップ６０１において、特徴文字列抽出プログラム１１６を起動し、ワークエリア１３０に格納されている同一文字種文字列から特徴文字列を抽出し、同じくワークエリア１３０に格納する。

【００９０】そして、ステップ６０２において、出現頻度ファイル作成プログラム１１７を起動し、ワークエリ

ア１３０に格納されている特徴文字列を参照して、その出現頻度を計数し、出現頻度ファイル１０４として磁気ディスク装置１０５に格納する。

【００９１】以上が、出現頻度ファイル作成登録プログラム１１４の処理手順である。

【００９２】次に、図６に示したステップ６０１において出現頻度ファイル作成登録プログラム１１４により起動される特徴文字列抽出プログラム１１６の処理手順について、図７のＰＡＤ図を用いて説明する。

【００９３】特徴文字列抽出プログラム１１６は、同一文字種文字列抽出プログラム１１５により抽出された同一文字種文字列の数を調べ、全ての同一文字種文字列についてステップ７０１以降を繰り返し実行する（ステップ７００）。

【００９４】ステップ７０１では、ワークエリア１３０に格納されている同一文字種文字列の文字種を判定し、その文字種が漢字の場合にはステップ７０２を実行し、カタカナの場合には、ステップ７０３を実行する。

【００９５】ステップ７０２では、後述する漢字文字列対応特徴文字列抽出プログラム１２７を起動し、漢字文字列から特徴文字列を抽出する。

【００９６】ステップ７０３では、同様に後述するカタカナ文字列対応特徴文字列抽出プログラム１２８を起動し、カタカナ文字列から特徴文字列を抽出する。

【００９７】以上が、特徴文字列抽出プログラム１１６の処理手順である。

【００９８】次に、図７に示したステップ７０２で特徴文字列抽出プログラム１１６により起動される漢字文字列対応特徴文字列抽出プログラム１２７の処理手順について、図８のＰＡＤ図を用いて説明する。

【００９９】漢字文字列対応特徴文字列抽出プログラム１２７では、ステップ８００において、同一文字種文字列抽出プログラム１１５により抽出されワークエリア１３０に格納されている漢字文字列を取得する。　そして、ステップ８０１において、上記ステップ８００で取得した漢字文字列の先頭から一文字ずつずらしながら、n-gram（nの値は、予め定めておく）を特徴文字列として抽出する。

【０１００】以上が、漢字文字列対応特徴文字列抽出プログラム１２７の処理手順である。

【０１０１】次に、図７に示したステップ７０３で特徴文字列抽出プログラム１１６により起動されるカタカナ文字列対応特徴文字列抽出プログラム１２８の処理手順について、図９のＰＡＤ図を用いて説明する。

【０１０２】カタカナ文字列対応特徴文字列抽出プログラム１２８では、ステップ９００において、同一文字種文字列抽出プログラム１１５により抽出されワークエリア１３０に格納されているカタカナ文字列を取得する。

【０１０３】そして、ステップ９０１において、上記ステップ９００で取得したカタカナ文字列そのものを特徴文字列として抽出する。

【０１０４】以上が、カタカナ文字列対応特徴文字列抽出プログラム１２８の処理手順である。

【０１０５】以下に、図７に示した特徴文字列抽出プログラム１１６の処理手順について具体例を用いて説明する。

【０１０６】まず、図７の特徴文字列抽出プログラム１１６のステップ７０２における漢字文字列対応特徴文字列抽出プログラム１２７と、ステップ７０３におけるカタカナ文字列対応特徴文字列抽出プログラム１２８の処理手順について、図１０～図１２の例を用いて説明する。　漢字文字列対応特徴文字列抽出プログラム１２７とカタカナ文字列対応特徴文字列抽出プログラム１２８は特徴文字列抽出プログラム１１６によって起動される。このとき、同一文字種文字列抽出プログラム１１５によって抽出された同一文字種文字列が漢字文字列対応特徴文字列抽出プログラム１２７とカタカナ文字列対応特徴文字列抽出プログラム１２８へワークエリア１３０を介して渡される。

【０１０７】図１０は文書１、文書２、文書３および文書４からなるテキスト１０３から、同一文字種文字列抽出プログラム１１５により同一文字種文字列が抽出された結果を示したものである。例えば、文書２「新しいソフトウェアの開発作業」からは「新」「しい」「ソフトウェア」「の」「開発作業」という５個の同一文字種文字列が抽出される。

【０１０８】この抽出された同一文字種文字列の文字種にしたがって、特徴文字列抽出プログラム１１６は、漢字文字列対応特徴文字列抽出プログラム１２７あるいはカタカナ文字列対応特徴文字列抽出プログラム１２８を起動する。

【０１０９】漢字文字列対応特徴文字列抽出プログラム１２７は、ワークエリア１３０に格納されている漢字文字列の先頭から一文字ずつずらしながら、全ての2-gramを特徴文字列として抽出する。図１１は、図１０の例で抽出された漢字文字列から、漢字文字列対応特徴文字列抽出プログラム１２７により特徴文字列を抽出した結果を示している。例えば、同一文字種文字列１０００の中で文書２から抽出された「新」「しい」「ソフトウェア」「の」「開発作業」からは、「開発」「発作」「作業」が抽出される。

【０１１０】カタカナ文字列対応特徴文字列抽出プログラム１２８は、ワークエリア１３０に格納されているカタカナ文字列そのものを特徴文字列として抽出する。図１２は、図１０の例で抽出されたカタカナ文字列から、カタカナ文字列対応特徴文字列抽出プログラムにより特徴文字列を抽出した結果である。例えば、同一文字種文字列１０００の中で文書２から抽出された「新」「しい」「ソフトウェア」「の」「開発作業」からは、「ソフトウェア」が抽出される。

【０１１１】以上が、第一の実施例における特徴文字列抽出プログラム１１６のステップ７０２における漢字文字列対応特徴文字列抽出プログラム１２７と、ステップ７０３におけるカタカナ文字列対応特徴文字列抽出プログラム１２８の処理手順である。

【０１１２】この例では、漢字文字列対応特徴文字列抽出プログラム１２７の処理として、漢字文字列から2-gramを特徴文字列として抽出するものとして説明したが、1-gram、あるいは3-gram以上であっても、さらには、それらの組み合わせであっても、同様に特徴文字列抽出の処理を行うことができることは明らかであろう。

【０１１３】次に、図４に示したステップ４０４でシステム制御プログラム１１０により起動される検索制御プログラム１１８による類似文書検索の処理手順について、図１３のＰＡＤ図を用いて説明する。

【０１１４】検索制御プログラム１１８は、まずステップ１３００で検索条件式解析プログラム１１９を起動し、キーボード１０１から入力された検索条件式を解析し、検索条件式のパラメータとして指定された種文書番号を抽出する。

【０１１５】次に、ステップ１３０１で類似文書検索プログラム１２０を起動し、上記ステップ１３００で抽出された種文書番号に対し、磁気ディスク装置１０５に格納されているテキスト１０３中の各文書の類似度を算出する。

【０１１６】そして、ステップ１３０２において、類似度ソートプログラム１２６を起動し、上記ステップ１３０１で算出された各文書の類似度を降順にソートする。

【０１１７】最後に、ステップ１３０３において上記ステップ１３０２でソートされた類似度を各文書番号と共に出力する。

【０１１８】以上が、検索制御プログラム１１８による文書検索の処理手順である。

【０１１９】次に、図１３に示したステップ１３０１で検索制御プログラム１１８により起動される類似文書検索プログラム１２０の処理手順について、図１４のＰＡＤ図を用いて説明する。　類似文書検索プログラム１２０は、まずステップ１４００で種文書読込みプログラム１２１を起動し、検索条件式解析プログラム１１９によって検索条件式から抽出された文書番号の種文書をワークエリア１３０に読み込む。ここで、種文書は、テキスト１０３中に格納されている文書を読み込むだけでなく、フロッピディスク１０７、ＣＤ－ＲＯＭ装置（図１には示していない）や通信回線等を用いて、他の装置から入力するような構成を取ることも可能であり、また、全文検索システム等による検索結果から入力するような構成を取ることも可能であり、類似度ソートプログラム１２６の出力から種文書を選択する構成を取ることも可能である。

【０１２０】次に、ステップ１４０１において、同一文字種文字列抽出プログラム１１５を起動し、上記種文書読込みステップ１４００で読み込んだ種文書のテキストを、文字種境界で分割して同一文字種文字列を取得し、ワークエリア１３０に格納する。

【０１２１】そして、ステップ１４０２において特徴文字列抽出プログラム１１６を起動し、上記同一文字種文字列抽出ステップ１４０１で取得した同一文字種文字列から、特徴文字列を抽出する。

【０１２２】図１５に、この処理の具体例を示す。特徴文字列抽出プログラム１１６の処理手順に関しては、前に説明した通りである。

【０１２３】本例では、種文書である文書２「新しいソフトウェアの開発作業」から、「新」「しい」「ソフトウェア」「の」「開発作業」という５個の同一文字種文字列１５００が抽出されることになる。この抽出された同一文字種文字列１５００の文字種にしたがって、特徴文字列を抽出する。この結果、文書２からは「ソフトウェア」「開発」「発作」「作業」の４個の特徴文字列１５０１が抽出される。

【０１２４】次に図１４のステップ１４０３で、出現頻度計数プログラム１２３を起動し、上記特徴文字列抽出ステップ１４０２で抽出した特徴文字列の種文書内における出現頻度を計数する。

【０１２５】図１６に、この具体例を示す。本図は、図１５に例示した種文書から抽出された特徴文字列１５０１の出現頻度を計数した結果を示している。すなわち、「（ソフトウェア，１）、（開発，１）、（発作，１）、（作業，１）」という出現頻度１６００が得られている。ここで、例えば（開発，１）は、特徴文字列「開発」が「１」回出現するということを示している。

【０１２６】次に、図１４のステップ１４０４で、出現頻度ファイル読込みプログラム１２４を起動し、上記特徴文字列抽出ステップ１４０２で抽出した特徴文字列の、テキスト１０３中の各文書における出現頻度を出現頻度ファイル１０４から読み込む。

【０１２７】図１７に、この具体例を示す。ここでは、図１５の例で抽出された特徴文字列１５０１のテキスト１０３中の各文書における出現頻度を、読み込んだ出現頻度ファイルから取得した結果を示している。

【０１２８】この例では、種文書から抽出された「ソフトウェア」「開発」「発作」「作業」という４個の特徴文字列１５０１の出現頻度を、出現頻度ファイル１０４から得る。この結果、出現頻度１７００として、例えば文書３の場合「（ソフトウェア，１）、（開発，１）、（発作，０）、（作業，０）」という値を得ることができる。

【０１２９】最後に、図１４のステップ１４０５で、類似度算出プログラム１２５を起動し、上記出現頻度計数ステップ１４０３で計数した特徴文字列の種文書内における出現頻度と、上記出現頻度ファイル読込みステップ

１４０４で読み込んだ特徴文字列のテキスト１０３内の各文書における出現頻度から、テキスト１０３中の各文書との類似度を算出する。

【０１３０】図１８に、この具体例を示す。ここでは、図１６の例で計数した種文書における出現頻度１６００および図１７の例で取得したテキスト１０３中の各文書における出現頻度１７００を用いて、各文書の類似度S(1)～S(4)を算出した結果を示している。すなわち、次のような結果が得られる。

【０１３１】

S(1)=0.077

S(2)=1.0

S(3)=0.263

S(4)=0.148

本実施例では、この類似度の算出に、従来技術２に開示されている式（１）を用いるが、他の方法を用いても構わない。

【０１３２】以上が、類似文書検索プログラム１１８の処理手順である。

【０１３３】以上が、本発明の第一の実施例である。

【０１３４】なお、本実施例においては、特徴文字列抽出プログラム１１６は、漢字対応特徴文字列抽出プログラム１２７およびカタカナ文字列対応特徴文字列抽出プログラム１２８を含む構成としたが、英字や数字等に対応した特徴文字列抽出プログラムを含む構成としてもよいし、漢字文字列対応特徴文字列抽出プログラム１２７あるいはカタカナ文字列対応特徴文字列抽出プログラム１２８を含まない構成であってもよい。

【０１３５】また、本実施例においては、同一文字種文字列から特徴文字列を抽出する構成としたが、特定の文字種間を境界として前後に跨る部分文字列を特徴文字列として抽出することにより、例えば、「Ｆ１」や「ビタミンＣ」等の文字列を検索に用いることもでき、さらに高精度な類似文書検索を実現することが可能となる。

【０１３６】さらに、本実施例においては、出現頻度ファイル１０４を図２に示した表形式で作成されるものとしたが、この方法では、データベースが大規模になるに伴い特徴文字列の種類が増加するため、出現頻度ファイル読込みステップ１４０４の処理に長大な時間を要することになる。この問題は、特徴文字列に対して検索用のインデクスを付加することにより解決できる。これにより、大規模なデータベースに対しても高速な類似文書検索を実現することが可能となる。この特徴文字列に対する検索用インデクスとしては、「特開平８－３２９１１２号公報」等に開示されているような単語インデクス方式を用いることができる。

【０１３７】次に、本発明の第二の実施例について図１９を用いて説明する。

【０１３８】本発明を適用した類似文書検索システムの第二例は、種文書から抽出した特徴文字列のデータベース内の各文書における出現頻度の取得に、検索漏れのない全文検索インデクスを利用するものである。これにより、本類似文書検索システムを全文検索システムと組み合わせて実現した場合に、出現頻度ファイルをもつ必要がなくなる。

【０１３９】すなわち、本方法によれば、第一の実施例における出現頻度ファイル１０４の特徴文字列の検索に全文検索インデクスを利用することができ、大規模なデータベースに対しても高速な類似文書検索を実現することが可能となる。さらに、出現頻度ファイル１０４を全文検索用インデクスで代用するため、第一の実施例に比べ必要となる磁気ディスク容量を削減できることになる。

【０１４０】本実施例は、第一の実施例（図１）とほぼ同様の構成を取るが、類似文書検索プログラム１２０を構成する出現頻度ファイル読込みプログラム１２４が異なる。このプログラムの代わりに、図１９に示すように、特徴文字列検索プログラム１９００が用いられる。

【０１４１】以下、本実施例における処理手順のうち、第一の実施例とは異なる類似文書検索プログラム１２０ａの処理手順について図２０を用いて説明する。

【０１４２】ここで、第一の実施例における類似文書検索プログラム１２０（図１４）と異なる点は、出現頻度取得ステップ２００４だけである。他の処理ステップの処理手順は、第一の実施例で説明した通りである。

【０１４３】出現頻度取得ステップ２００４では、特徴文字列検索プログラム１９００を起動し、特徴文字列抽出ステップ１４０２で抽出された特徴文字列を全文検索システム１９０１で検索することにより、テキスト１０３内の各文書における出現頻度を取得する。

【０１４４】本実施例の出現頻度取得ステップ２００４で用いる特徴文字列検索プログラム１９００は、検索漏れがなく、かつ、各文書における出現頻度を取得できる全文検索方式であれば、どのような方式を適用しても構わない。例えば、「特開昭６４－３５６２７号公報」（以下、従来技術３と呼ぶ）で開示されているようなn-gramインデクス方式を用いることも可能である。

【０１４５】この従来技術３によるn-gramインデクス方式では、文書の登録時に、データベースへ登録する文書のテキストデータからn-gramとそのn-gramのテキスト中における出現位置を抽出し、全文検索用インデクス１９０３として磁気ディスク装置１９０２に格納しておく。検索時には指定された検索ターム中に出現するｎ－ｇｒａｍを抽出し、これらに対応するインデクスを上記磁気ディスク装置１９０２から読み込み、インデクス中のｎ－ｇｒａｍの出現位置を比較し、検索タームから抽出したn-gramの位置関係とインデクス中のn-gramの位置関係が等しいかどうかを判定することによって、指定された検索タームが出現する文書を高速に検索する。

【０１４６】この方式を用いて、特徴文字列を検索ター

ムとして全文検索システム１９０１へ入力し、該検索タームの出現文書とその位置情報を取得することにより、該特徴文字列の各文書における出現頻度を求めることが可能となる。

【０１４７】以下、この従来技術３を用いた出現頻度の算出方法を図２１を用いて具体的に説明する。なお本図では、n-gramのnの値を２としている。

【０１４８】まず、文書の登録時にデータベースに登録するテキスト２１０１がインデクス作成部２１０２に読み込まれ、n-gramインデクス２１００が作成される。このn-gramインデクス２１００には、テキスト２１０１に出現する全ての2-gramとテキスト２１０１におけるその2-gramの出現位置が格納される。

【０１４９】本図に示すテキスト２１０１では、「心電」という2-gramはテキスト２１０１（文書番号「１」）の５文字目、１５文字目、･･･に現われるので、n-gramインデクス２１００には2-gram「心電」とこれに対応したかたちで出現位置｛（１，５）、（１，１５）、･･･｝が格納される。

【０１５０】検索時には、まず、検索タームがn-gram抽出部２１０３に入力され、検索ターム中に出現する全てのn-gramとそのn-gramの検索タームにおける出現位置が抽出される。次に、抽出されたn-gramとこれに対応するn-gramの検索タームにおける出現位置がインデクス検索部２１０４に入力される。

【０１５１】インデクス検索部２１０４では、検索タームから抽出されたn-gramに対応するインデクスがn-gramインデクス２１００から読み込まれ、これらのインデクスの中から文書番号が一致し、かつ検索ターム中の位置関係と同じ位置関係を持つものが抽出され、検索結果として出力される。

【０１５２】検索タームとして「心電図」が入力された本図の場合、まず、n-gram抽出部２１０３において、（n-gram「心電」、n-gram位置「１」）と（n-gram「電図」、n-gram位置「２」）が抽出される。ここで、 n-gram位置「１」は検索タームの先頭、 n-gram位置「２」はその次の文字位置を示す。

【０１５３】次に、インデクス検索部２１０４において、n-gramインデクス２１００からn-gram「心電」と「電図」に対応するインデクスが読み込まれる。これらのインデクスにおける出現位置がn-gram位置「１」とn-gram位置「２」のように連続するものが、すなわち隣接するものが抽出され検索結果として出力される。

【０１５４】本図では、 n-gram「心電」の出現位置「１５」とn-gram「電図」の出現位置「１６」が隣接するため、 n-gram「心電図」が文字列として存在することが分かり、文書１中に検索ターム「心電図」が出現することが示される。しかし、 n-gram「心電」の出現位置「５」とn-gram「電図」の出現位置「１６」は隣接していないため、この位置には検索ターム「心電図」が出現しないことが分かる。

【０１５５】本方法において、検索タームとして特徴文字列入力した場合、上記インデクス検索部２１０４から検索結果として出力される出現位置を計数することにより、該当特徴文字列の出現頻度を得ることが可能となる。

【０１５６】以上説明したように、本実施例によれば、出現頻度ファイルの特徴文字列検索用インデクスと出現頻度ファイルの代わりに、全文検索インデクスを利用できるため、大規模なデータベースに対しても余分なファイルを増やさずに、高速に類似文書検索を実現することが可能となる。

【０１５７】次に、本発明の第三の実施例について図２２を用いて説明する。

【０１５８】本発明を適用した類似文書検索システムの第三例は、種文書から抽出した特徴文字列の重要度を算出し、この重要度が所定値を満たす特徴文字列に限定して、データベース内の各文書における出現頻度を取得し、これに基づいて類似度を算出するものである。

【０１５９】すなわち、本方法は、第一の実施例における出現頻度ファイル読込みステップ１４０４で出現頻度の取得対象とする特徴文字列数を削減することによって、類似度算出に用いる特徴文字列数を削減し、文字数の多い種文書に対しても高速な類似文書検索を実現できるようにするものである。

【０１６０】本実施例は、第一の実施例（図１）とほぼ同様の構成を取るが、類似文書検索プログラム１２０が異なり、図２２に示すように、特徴文字列選択プログラム２２００を有する。

【０１６１】以下、本実施例における処理手順のうち、第一の実施例とは異なる類似文書検索プログラム１２０ｂの処理手順について図２３のＰＡＤ図を用いて説明する。

【０１６２】ここで、第一の実施例における類似文書検索プログラム１２０（図１４）の処理手順と異なる点は、特徴文字列選択ステップ２３００だけである。他の処理ステップの処理手順は、第一の実施例で説明した通りである。

【０１６３】特徴文字列選択ステップ２３００では、特徴文字列選択プログラム２２００を起動し、特徴文字列抽出ステップ１４０２（特徴文字列抽出プログラム１１６）で抽出した特徴文字列の重要度を算出し、所定の値を満たす文字列を類似検索用の特徴文字列として選択する。

【０１６４】以下、特徴文字列選択ステップ２３００で起動される特徴文字列選択プログラム２２００の処理手順を図２４のＰＡＤ図を用いて説明する。

【０１６５】特徴文字列選択プログラム２２００は、まず、ステップ２４００において特徴文字列抽出ステップ１４０２で抽出された特徴文字列を取得すし、ワークエ

リア１３０に格納する。

【０１６６】次に、ステップ２４０１で各特徴文字列が出現する文書数を出現頻度ファイル１０４から取得する。

【０１６７】そして、ステップ２４０２において、所定の重要度算出式を用いて該特徴文字列の重要度を算出する。

【０１６８】この結果、該重要度が所定値を満たす特徴文字列に限定し、これを類似度算出用の特徴文字列として抽出する（ステップ２４０３）。この重要度には、従来技術２の共通性ウェイトを用いてもよい。本実施例では、重要度の算出に以下に示す式（２）を用いる。

【０１６９】

【数２】

数２

$$\text{特徴文字列の重要度} = 1 + \log_2 \frac{n}{NumDoc} \quad \cdots \quad 2$$

【０１７０】ここで、nはデータベース中の文書数、NumDocは特徴文字列のデータベースにおける出現文書数を示す。この値は、特徴文字列がデータベース中の全ての文書に出現する場合に最も小さく、特定の文書に偏って出現する場合に大きくなる。

【０１７１】また、特徴文字列を抽出する際に基準とする閾値としては、上限とする重要度と下限とする重要度を予め定めておいてもよいし、重要度の上位k個（kは1以上の予め定められた整数）を採るものとしてもよい。

【０１７２】以下、図２５に示す具体例で特徴文字列選択ステップ２２００の処理手順を説明する。なお本図では、図１５の例で抽出した特徴文字列１５０１を対象とし、重要度が3.0以上である特徴文字列を選択するものとする。

【０１７３】まず、ステップ１４０４（図２３）でワークエリア１３０に読み込んだ出現頻度ファイル１０４から各特徴文字列の出現文書数を取得する。この例では、文書２の特徴文字列１５０１の各出現文書数２５００として、［ソフトウェア，２］、［開発，３］、［発作，２］、［作業，２］が得られる。ここで、［ソフトウェア，２］は、特徴文字列「ソフトウェア」がデータベース中の「２」つの文書に出現することを表わす。

【０１７４】次に、各特徴文字列の出現文書数２５００から重要度２５０１を算出し、重要度が3.0以上の特徴文字列を抽出する。この結果、「ソフトウェア」という１個の特徴文字列２５０２が類似度算出用の特徴文字列として選択されることになる。

【０１７５】このように、特徴文字列の個数を４個から１個に削減することができるため、類似度算出に要する時間を大幅に削減することができる。

【０１７６】なお、本実施例では、出現頻度ファイル１０４を参照して、各特徴文字列の出現文書数を取得する構成としたが、文書登録時に各文書中の特徴文字列を計数し、各特徴文字列の出現文書数を求め、これを出現文書数ファイルとして記憶しておくことにより、さらに高速に特徴文字列を選択することも可能である。

【０１７７】また、本実施例では、出現頻度ファイル１０４を参照して、各特徴文字列の出現文書数を取得し重要度を算出する構成としたが、文書登録時に各文書における特徴文字列の重要度を算出し、これを重要度ファイルとして記憶しておくことにより、さらに高速に特徴文字列を選択することが可能となる。

【０１７８】さらに、本実施例では、重要度の算出に特徴文字列のデータベース中の出現文書数を用いたが、例えば、特徴文字列の文字種類や文字列長、種文書内の出現頻度あるいは出現位置等の情報のいずれか一つ、あるいは、それらを組み合わせることにより算出することも可能である。

【０１７９】以上説明したように、本発明によれば、分かち書きのない日本語のような文書に対して、類似文書検索を行なった場合においても、種文書から文字列を機械的に抽出することにより、どんな単語についても漏れのない検索を行なうことが可能となる。また、文字種に応じて特徴文字列を抽出することにより、意味のまとまった文字列を用いて検索を行なうことができるため、高精度な類似文書検索を実現することができるようになる。さらに、抽出する文字列の種類が大幅に削減されるため、高速に類似文書を検索することができるようになる。

【０１８０】さらに、全文検索システムと組み合わせて用いることにより、大規模な文書データベースに対しても、高速な類似文書検索が実現可能となる。

【０１８１】

【発明の効果】本発明によれば、単語辞書を用いずに類似文書検索を行なった場合でも、意味のまとまった文字列を用いて検索を行なうことができるため、高精度な類似文書検索を実現することができる。また、抽出する文字列の文字種に応じて最適な長さの部分文字列（n-gram）を抽出するため、高速に類似文書を検索することができるようになる。

【図面の簡単な説明】

【図１】本発明による類似文書検索システムの第一の実施例の全体構成を示す図である。

【図２】出現頻度ファイルの構成例を示す図である。

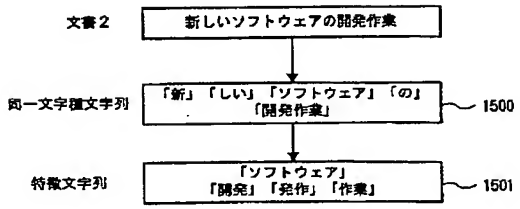
【図３】特徴文字列抽出処理の流れを示すＰＡＤ図である。

【図４】本発明の第一の実施例におけるシステム制御プログラムの処理手順を示すＰＡＤ図である。

【図５】本発明の第一の実施例における文書登録制御プログラムの処理手順を示すＰＡＤ図である。
【図６】本発明の第一の実施例における出現頻度ファイル作成プログラムの処理手順を示すＰＡＤ図である。
【図７】本発明の第一の実施例における特徴文字列抽出プログラムの処理手順を示すＰＡＤ図である。
【図８】本発明の第一の実施例における漢字文字列対応特徴文字列抽出プログラムの処理手順を示すＰＡＤ図である。
【図９】本発明の第一の実施例におけるカタカナ文字列対応特徴文字列抽出プログラムの処理手順を示すＰＡＤ図である。
【図１０】本発明の第一の実施例における同一文字種文字列抽出プログラムの処理例を示す図である。
【図１１】本発明の第一の実施例における漢字文字列対応特徴文字列抽出プログラムの処理例を示す図である。
【図１２】本発明の第一の実施例におけるカタカナ文字列対応特徴文字列抽出プログラムの処理例を示す図である。
【図１３】本発明の第一の実施例における検索制御プログラムの処理手順を示すＰＡＤ図である。
【図１４】本発明の第一の実施例における類似文書検索プログラムの処理手順を示すＰＡＤ図である。
【図１５】本発明の第一の実施例における特徴文字列抽出プログラムの処理例を示す図である。
【図１６】本発明の第一の実施例における出現頻度計数プログラムの処理例を示す図である。
【図１７】本発明の第一の実施例における出現頻度取得ファイル読込みプログラムの処理例を示す図である。
【図１８】本発明の第一の実施例における類似度算出プログラムの処理例を示す図である。
【図１９】本発明の第二の実施例における検索処理系のプログラム構成を示す図である。
【図２０】本発明の第二の実施例における類似文書検索プログラムの処理手順を示すＰＡＤ図である。
【図２１】本発明の第二の実施例におけるn-gramインデクスの例を示す図である。
【図２２】本発明の第三の実施例における検索処理系のプログラム構成を示す図である。
【図２３】本発明の第三の実施例における類似文書検索プログラムの処理手順を示すＰＡＤ図である。
【図２４】本発明の第三の実施例における特徴文字列選択プログラムの処理手順を示すＰＡＤ図である。
【図２５】本発明の第三の実施例における特徴文字列の選択の例を示す図である。
【符号の説明】
１００…ディスプレイ、
１０１…キーボード、
１０２…中央演算処理装置（ＣＰＵ）、
１０３…テキスト、
１０４…出現頻度ファイル、
１０５…磁気ディスク装置、
１０６…フロッピディスクドライブ（ＦＤＤ）、
１０７…フロッピディスク、
１０８…バス、
１０９…主メモリ、
１１０…システム制御プログラム、
１１１…文書登録制御プログラム、
１１２…共有ライブラリ、
１１３…テキスト登録プログラム、
１１４…出現頻度ファイル作成登録プログラム、
１１５…同一文字種文字列抽出プログラム、
１１６…登録用特徴文字列抽出プログラム、
１１７…出現頻度ファイル作成プログラム、
１１８…検索制御プログラム、
１１９…検索条件式解析プログラム、
１２０…類似文書検索プログラム、
１２１…種文書読込みプログラム、
１２３…出現頻度計数プログラム、
１２４…出現頻度読込みプログラム、
１２５…類似度算出プログラム、
１２６…類似度ソートプログラム、
１２７…漢字文字列対応特徴文字列抽出プログラム、
１２８…カタカナ文字列対応特徴文字列抽出プログラム、
１３０…ワークエリア

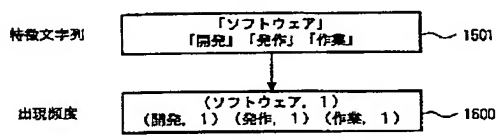
【図１５】

図１５



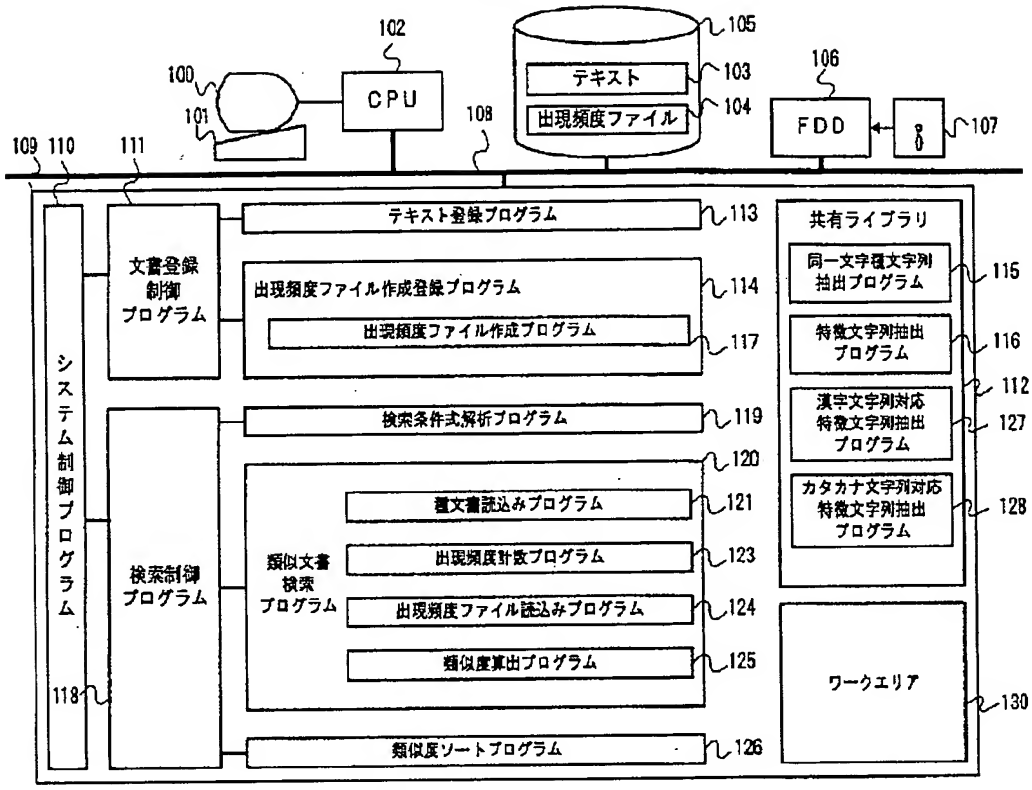
【図１６】

図１６



【図１】

図１



【図２】

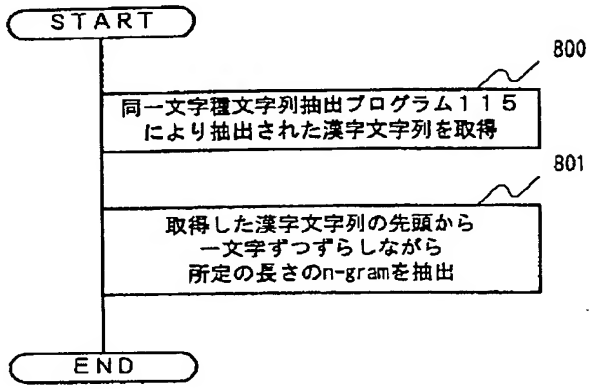
図２

No.	特徴文字列	文書1	文書2	文書3	文書4
1	新聞	1	0	0	0
2	開発	1	1	0	0
3	心電	2	0	0	0
4	電計	1	0	0	0
5	発作	1	0	0	0
6	作時	1	0	0	0
7	電図	1	0	0	0
8	ソフトウェア	0	0	1	0
9	作業	0	0	1	0
10	ソフト	0	0	1	0
11	支援	0	0	1	0
12	ソフトクリーム	0	0	0	1
13	配布	0	0	0	1
14	布作	0	0	0	1

200

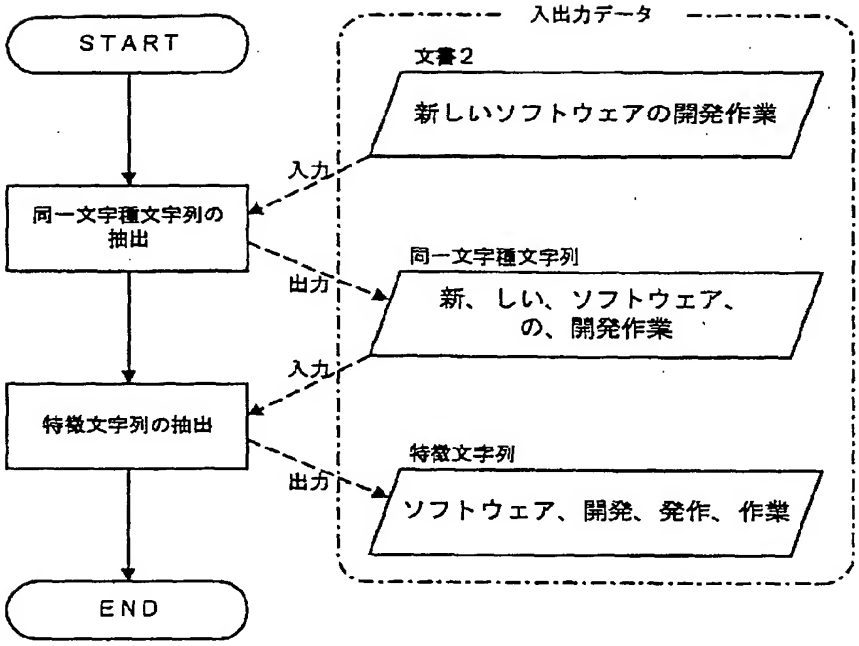
【図８】

図８



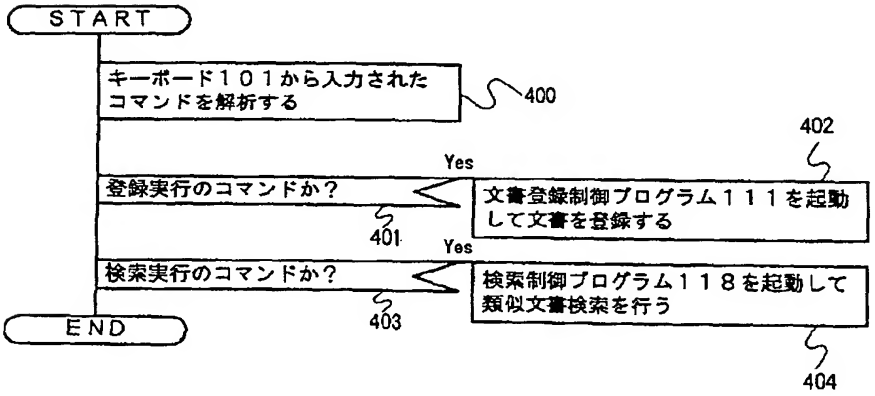
【図３】

図３


START
入出力データ
文書2
新しいソフトウェアの開発作業
入力
同一文字種文字列の抽出
出力
同一文字種文字列
新、しい、ソフトウェア、の、開発作業
入力
特徴文字列の抽出
出力
特徴文字列
ソフトウェア、開発、発作、作業
END

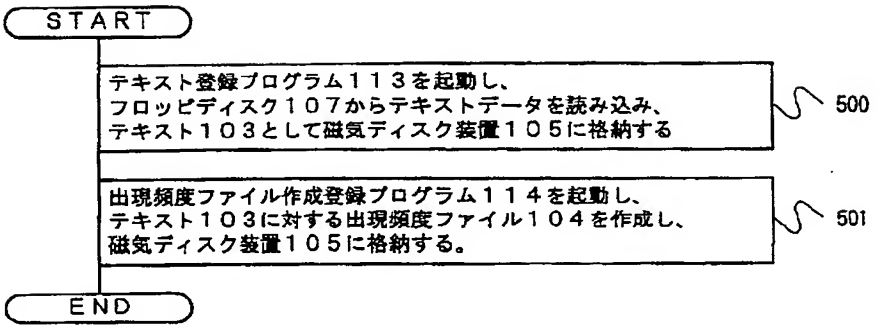
【図４】

図４


START
キーボード１０１から入力されたコマンドを解析する
400
402
Yes
登録実行のコマンドか？
文書登録制御プログラム１１１を起動して文書を登録する
401
Yes
検索実行のコマンドか？
検索制御プログラム１１８を起動して類似文書検索を行う
403
404
END

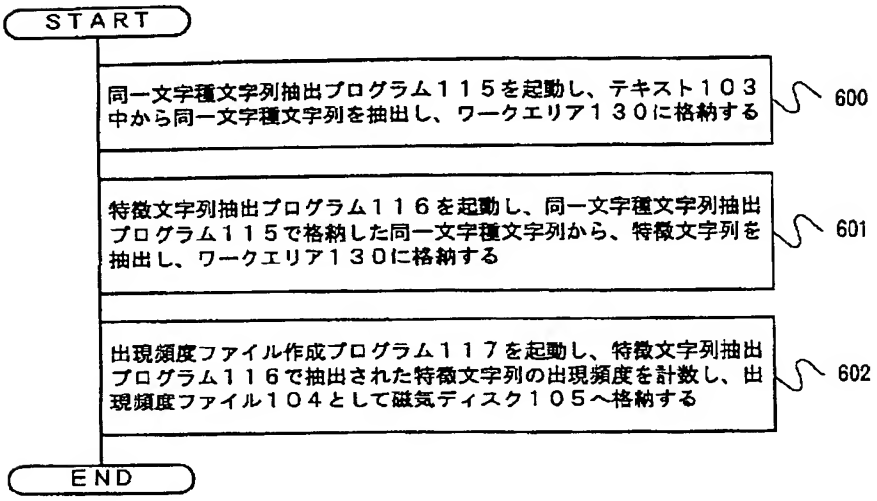
【図５】

図５



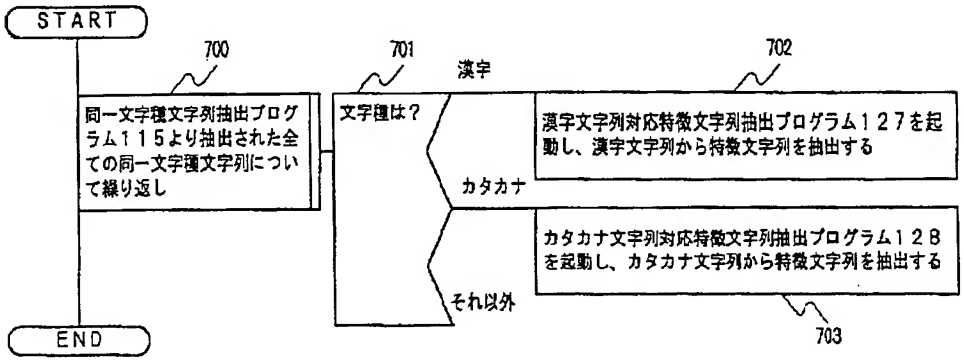
【図６】

図６



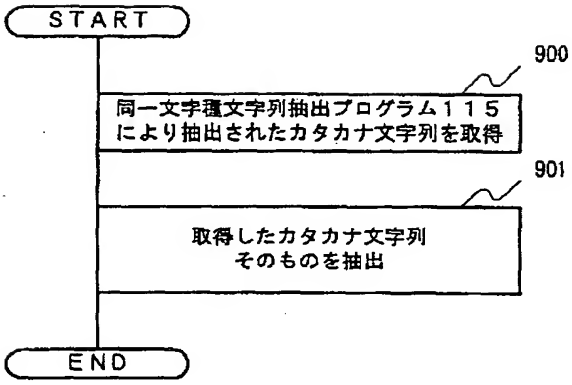
【図７】

図７



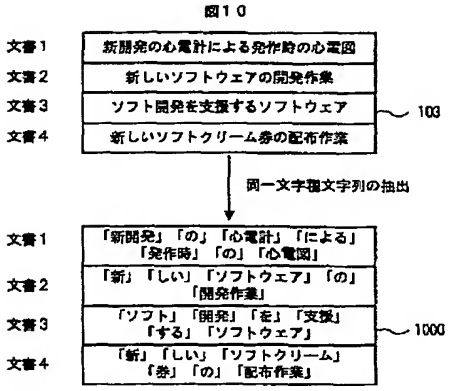
【図９】

図９



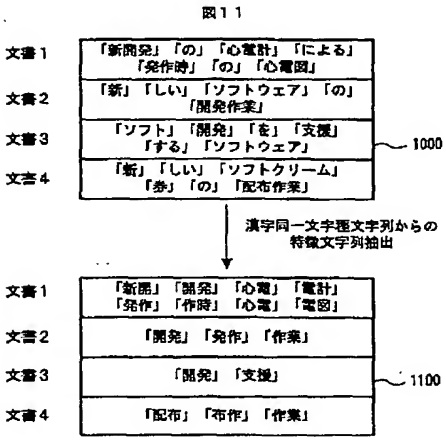
【図１０】

図１０



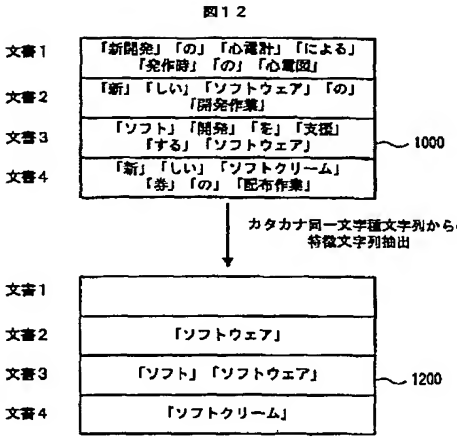
【図１１】

図１１



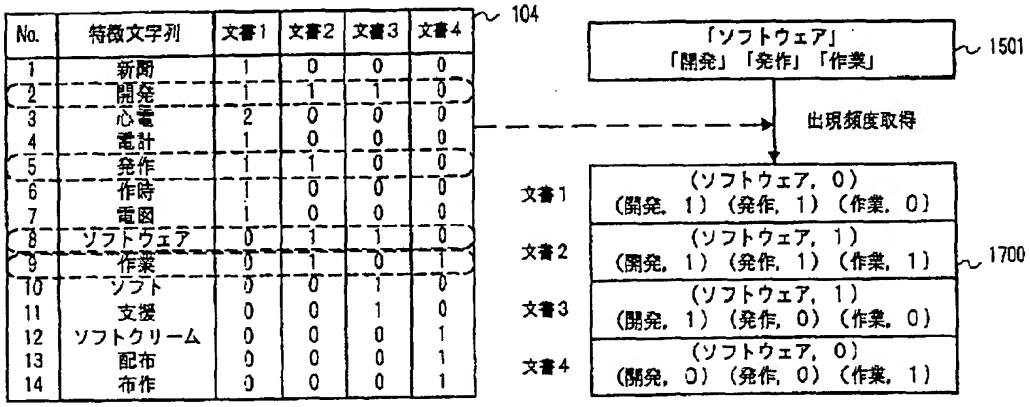
【図１２】

図１２



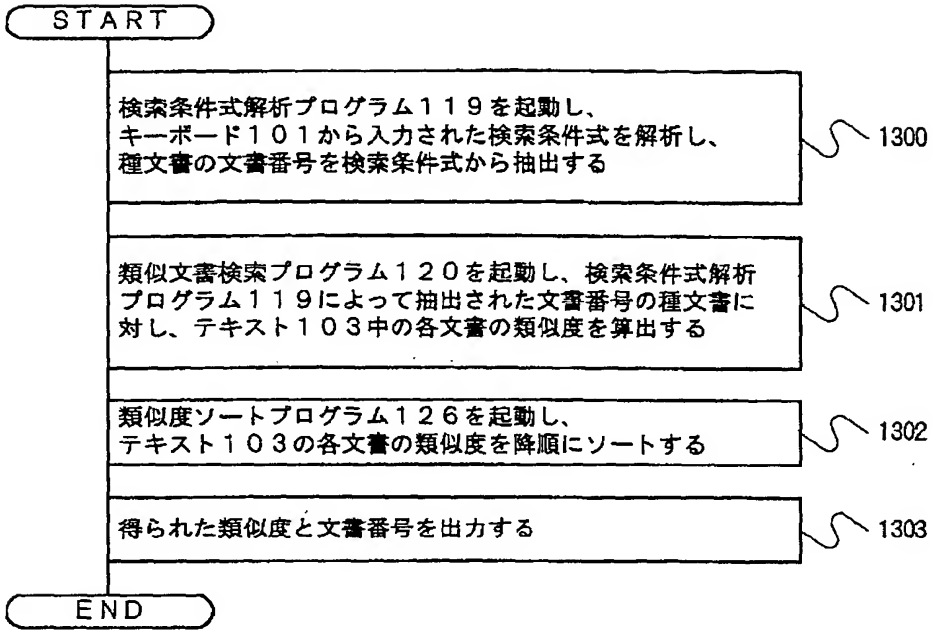
【図１７】

図１７



【図１３】

図１３



【図１８】

図１８

文書１の類似度

	ソフトウェア		開発		発作		作業	
	文書１	種文書	文書１	種文書	文書１	種文書	文書１	種文書
出現頻度	0	1	1	1	1	1	0	1
テキスト長	17	14	17	14	17	14	17	14
ウェイト	0	0.071	0.059	0.071	0.059	0.071	0	0.071
共通性ウェイト	0.033		0.048		0.033		0.033	
類似度	0.077							

文書２の類似度

	ソフトウェア		開発		発作		作業	
	文書２	種文書	文書２	種文書	文書２	種文書	文書２	種文書
出現頻度	1	1	1	1	1	1	1	1
テキスト長	14	14	14	14	14	14	14	14
ウェイト	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071
共通性ウェイト	0.033		0.048		0.033		0.033	
類似度	1.0							

文書３の類似度

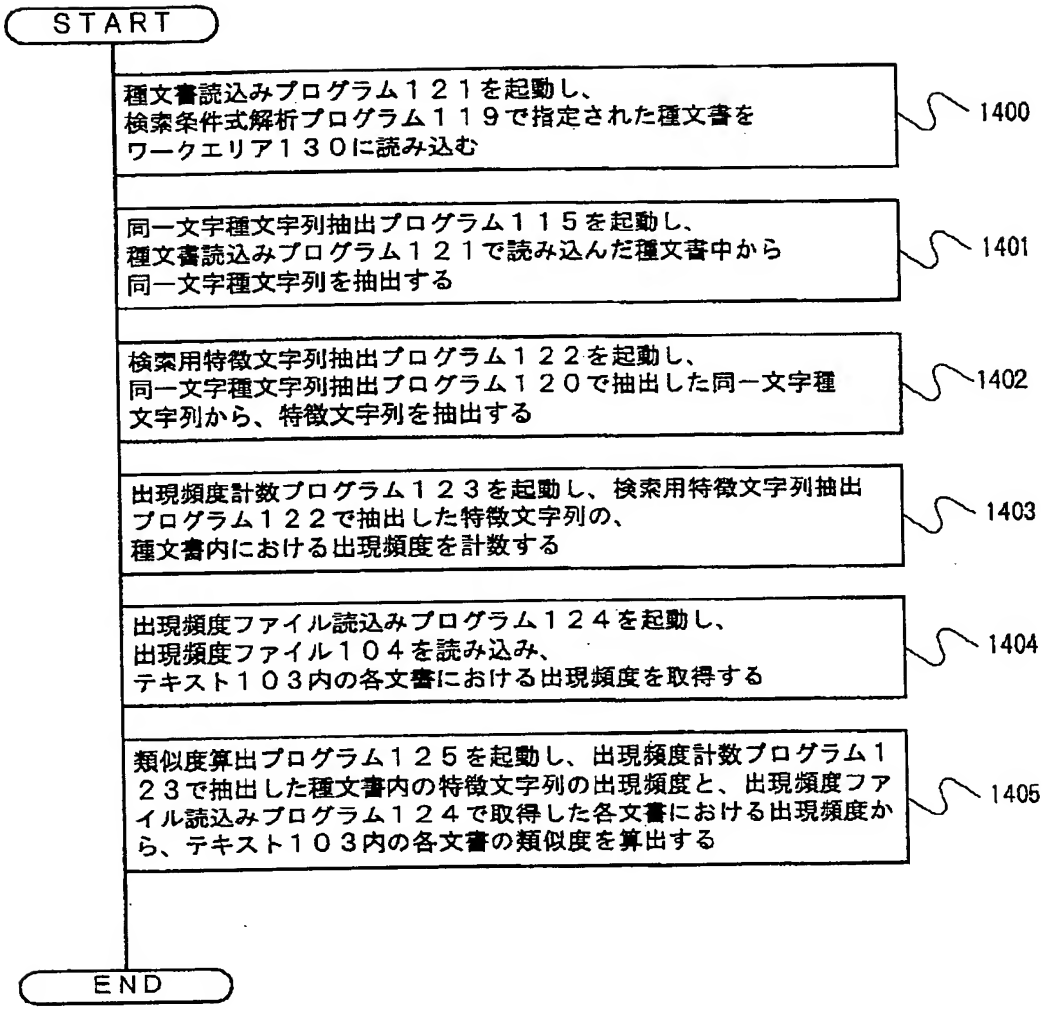
	ソフトウェア		開発		発作		作業	
	文書３	種文書	文書３	種文書	文書３	種文書	文書３	種文書
出現頻度	1	1	1	1	0	1	0	1
テキスト長	16	14	16	14	16	14	16	14
ウェイト	0.063	0.071	0.063	0.071	0	0.071	0	0.071
共通性ウェイト	0.033		0.048		0.033		0.033	
類似度	0.253							

文書４の類似度

	ソフトウェア		開発		発作		作業	
	文書４	種文書	文書４	種文書	文書４	種文書	文書４	種文書
出現頻度	0	1	0	1	0	1	1	1
テキスト長	16	14	16	14	16	14	16	14
ウェイト	0	0.071	0	0.071	0	0.071	0.063	0.071
共通性ウェイト	0.033		0.048		0.033		0.033	
類似度	0.148							

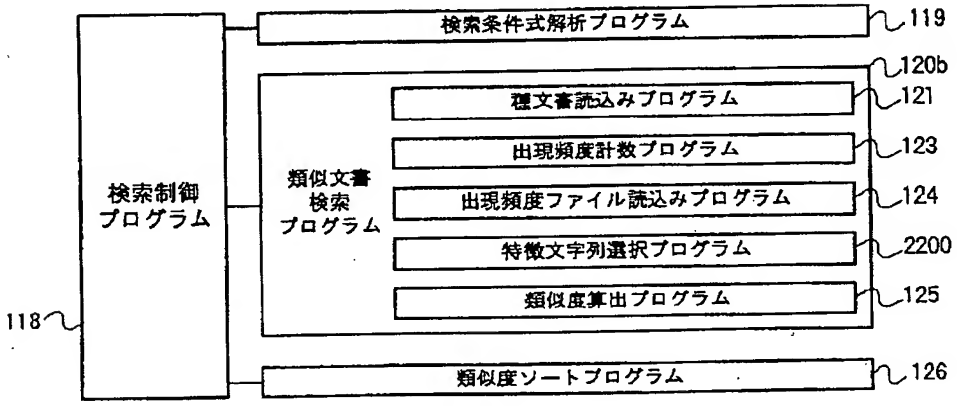
【図１４】

図１４



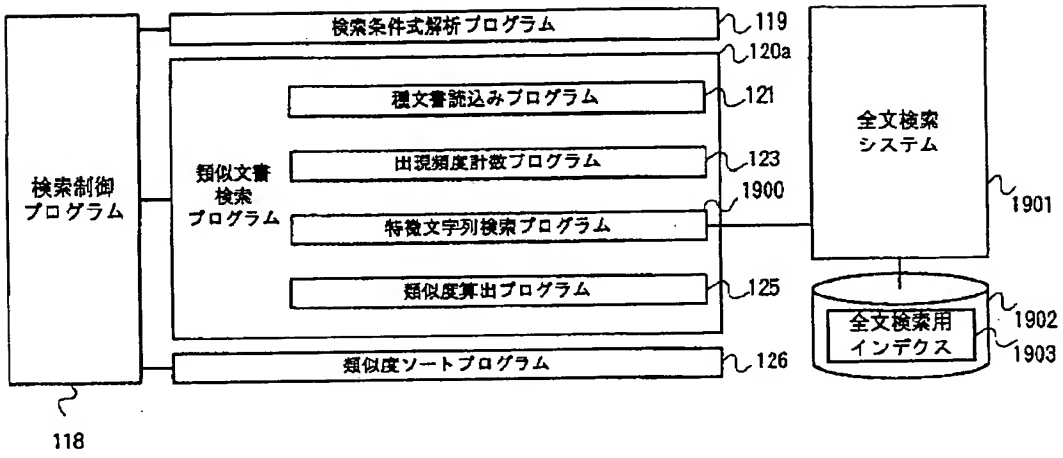
【図２２】

図２２



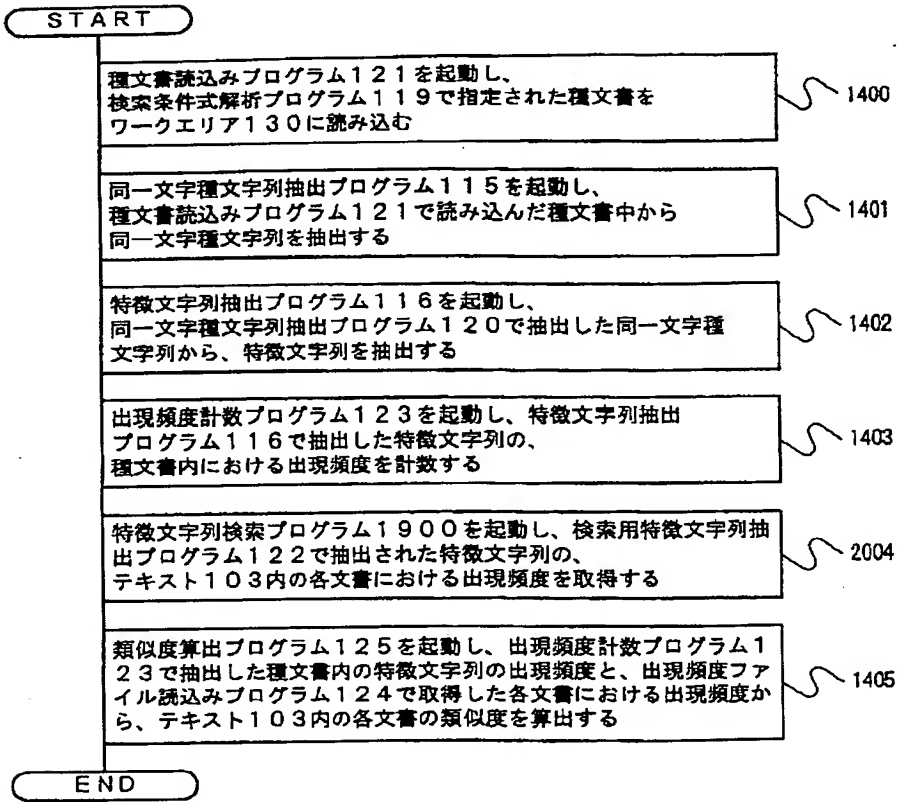
【図１９】

図１９



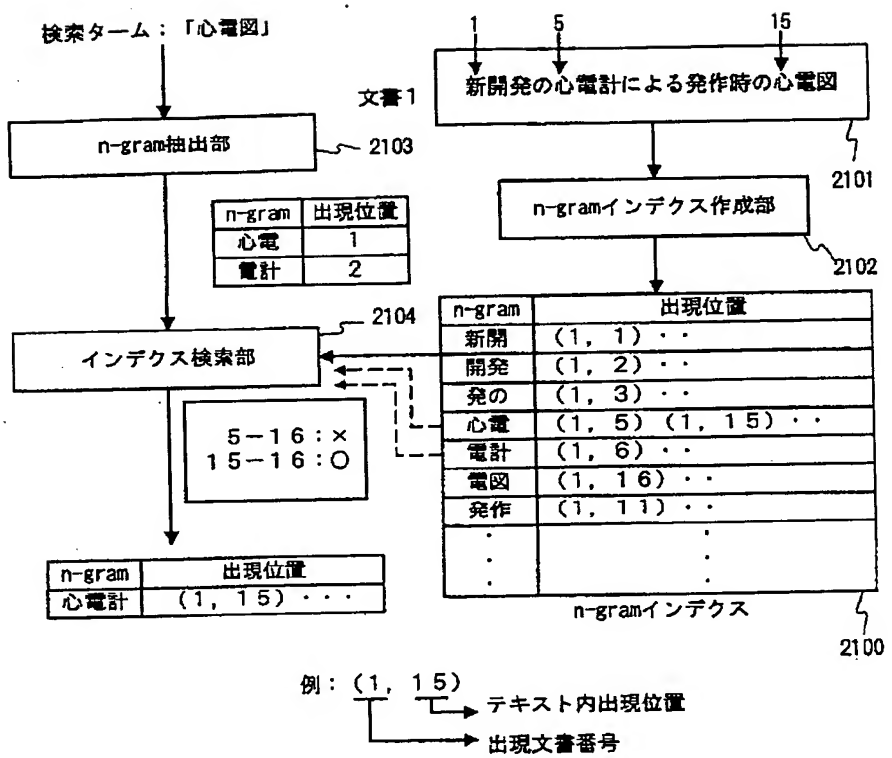
【図２０】

図２０



【図２１】

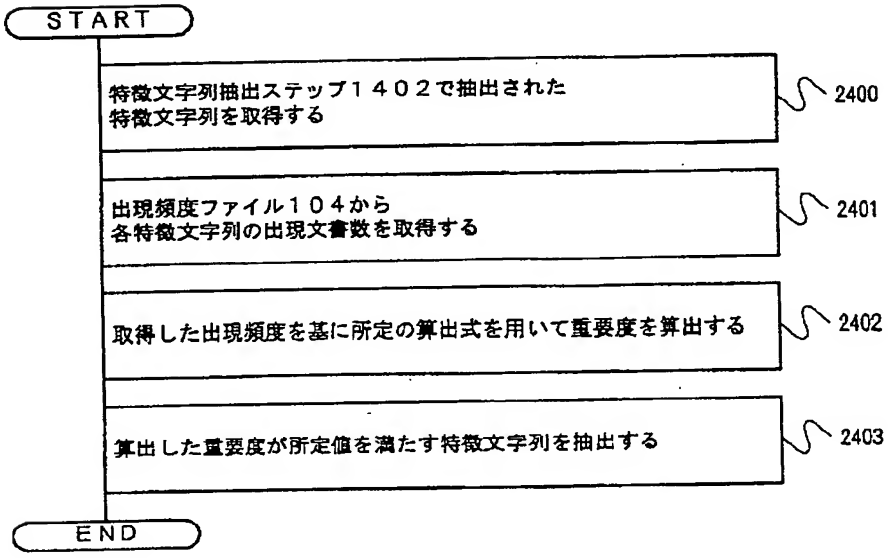
図２１



（n-gramインデクスの例）

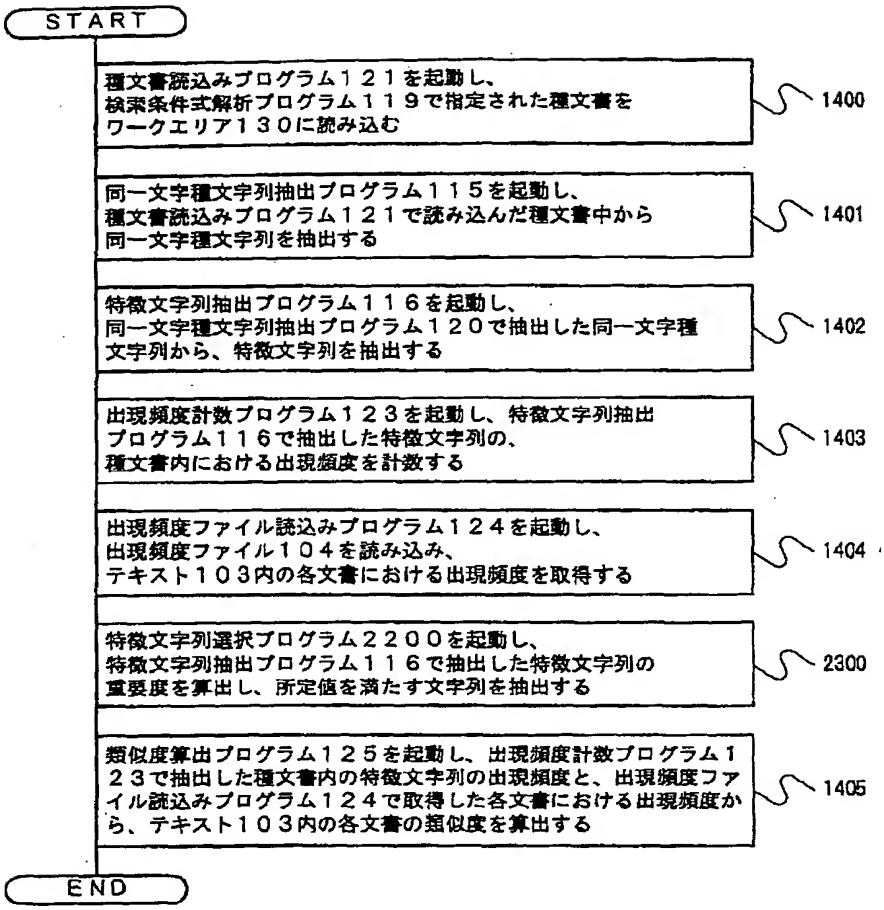
【図２４】

図２４



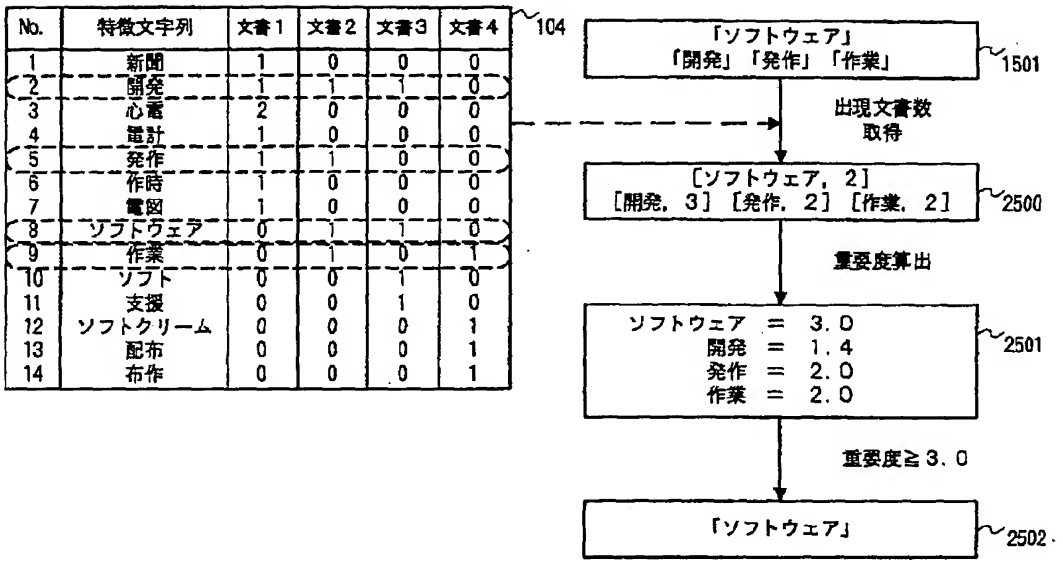
【図２３】

図２３


START
種文書読込みプログラム１２１を起動し、
検索条件式解析プログラム１１９で指定された種文書を
ワークエリア１３０に読み込む
1400
同一文字種文字列抽出プログラム１１５を起動し、
種文書読込みプログラム１２１で読み込んだ種文書中から
同一文字種文字列を抽出する
1401
特徴文字列抽出プログラム１１６を起動し、
同一文字種文字列抽出プログラム１２０で抽出した同一文字種
文字列から、特徴文字列を抽出する
1402
出現頻度計数プログラム１２３を起動し、特徴文字列抽出
プログラム１１６で抽出した特徴文字列の、
種文書内における出現頻度を計数する
1403
出現頻度ファイル読込みプログラム１２４を起動し、
出現頻度ファイル１０４を読み込み、
テキスト１０３内の各文書における出現頻度を取得する
1404
特徴文字列選択プログラム２２００を起動し、
特徴文字列抽出プログラム１１６で抽出した特徴文字列の
重要度を算出し、所定値を満たす文字列を抽出する
2300
類似度算出プログラム１２５を起動し、出現頻度計数プログラム１
２３で抽出した種文書内の特徴文字列の出現頻度と、出現頻度ファ
イル読込みプログラム１２４で取得した各文書における出現頻度か
ら、テキスト１０３内の各文書の類似度を算出する
1405
END

【図２５】

図２５



フロントページの続き

(72)発明者　菅谷　奈津子
神奈川県川崎市幸区鹿島田890番地　株式会社日立製作所情報・通信開発本部内

(72)発明者　川下　靖司
神奈川県横浜市戸塚区戸塚町3090番地　株式会社日立製作所ソフトウェア開発本部内

【公報種別】特許法第１７条の２の規定による補正の掲載
【部門区分】第６部門第３区分
【発行日】平成１４年１２月２０日（２００２．１２．２０）

【公開番号】特開平１１－１４３９０２
【公開日】平成１１年５月２８日（１９９９．５．２８）
【年通号数】公開特許公報１１－１４４０
【出願番号】特願平９－３０９０７８
【国際特許分類第７版】
　G06F　17/30
【ＦＩ】
　G06F　15/401　　310 A
　　15/403　　350 C

【手続補正書】
【提出日】平成１４年９月１７日（２００２．９．１７）
【手続補正１】
【補正対象書類名】明細書
【補正対象項目名】特許請求の範囲
【補正方法】変更
【補正内容】
【特許請求の範囲】
【請求項１】文字情報をコードデータとして蓄積したテキストデータベースを対象に、ユーザが指定した文書と類似する文書を検索する類似文書検索方法において、
ユーザが指定した文書のテキスト（指定テキストと呼ぶ）から所定の文字種の変わり目を境界として文字列を抽出する文字列抽出ステップと、
予め定められた一つ以上の文字列の種類に応じて、その中から一つ以上の部分文字列を抽出する検索用部分文字列抽出ステップと、
該指定テキストに対する該テキストデータベース中のテキストの類似度を所定の類似度算出式を用いて算出する類似度算出ステップを有することを特徴とした類似文書検索方法。
【請求項２】請求項１記載の類似文書検索方法における前記文字列抽出ステップは、
該指定テキストから抽出する文字列として、文字種の変わり目を境界として同一文字種からなる文字列を抽出する同一文字種文字列抽出ステップを有することを特徴とした類似文書検索方法。
【請求項３】請求項２記載の類似文書検索方法における前記検索用部分文字列抽出ステップは、
文字種に応じて予め定められた文字列長の部分文字列を検索用部分文字列として抽出する文字種別検索用部分文字列抽出ステップを有することを特徴とした類似文書検索方法。
【請求項４】請求項１、２および３に記載の類似文書検索方法における前記検索用部分文字列抽出ステップは、
予め定められた長さの文字列を検索用部分文字列として抽出する所定長文字列抽出ステップ、
前記文字列抽出ステップで抽出された文字列そのものを検索用部分文字列として抽出する最長文字列抽出ステップ、
前記文字列抽出ステップで抽出された文字列とその部分文字列の指定テキストにおける出現頻度比を算出し、所定値を満たす部分文字列を検索用部分文字列として抽出する高出現頻度比文字列抽出ステップ、
上記所定長文字列抽出ステップ、最長文字列抽出ステップおよび高出現頻度比文字列抽出ステップの中の少なくとも一つの抽出ステップで抽出された部分文字列から、予め作成しておいた、検索用部分文字列として抽出しない文字列を不要語として記載した排除文字列辞書に含まれる文字列を削除するステップ、および前記文字列抽出ステップで抽出された文字列から検索用部分文字列としては部分文字列を抽出しないステップ、
のいずれか一つ、あるいは、それらを組み合わせることにより検索用部分文字列を抽出する検索用部分文字列抽出ステップを有することを特徴とした類似文書検索方法。
【請求項５】請求項１、２、３および４に記載の類似文書検索方法はさらに、
前記検索用部分文字列抽出ステップで抽出された検索用部分文字列の重要度を、予め定められた算出式を用いて算出し、所定値を満たす検索用部分文字列を抽出する検索用部分文字列選択ステップを有することを特徴とした類似文書検索方法。
【請求項６】請求項５記載の類似文書検索方法における検索用部分文字列選択ステップは、
検索用部分文字列の抽出条件を設定する検索用部分文字列抽出条件設定ステップを有することを特徴とした類似文書検索方法。
【請求項７】請求項６記載の類似文書検索方法における検索用部分文字列抽出条件設定ステップは、

前記重要度の上限値を抽出条件として設定する重要度上限設定ステップ、
前記重要度の下限値を抽出条件として設定する重要度下限設定ステップ、
前記抽出する検索用部分文字列の個数を抽出条件として設定する検索用部分文字列抽出個数設定ステップのいずれか１つ、または、これらを組み合わせることにより検索用部分文字列を抽出することを特徴とした類似文書検索方法。